



TUGAS AKHIR - TE 141599

**RANCANG BANGUN *BUOYANT BOAT* BERTENAGA
SURYA UNTUK MEMBANTU KOMUNIKASI ANTARA
*REMOTELY OPERATED VEHICLE UNDERWATER
ROBOT* DENGAN *GROUND STATION***

Badrut Tamam
NRP 07111440000101

Dosen Pembimbing
Ronny Mardiyanto, ST., MT., Ph.D.

DEPATERMEN TEKNIK ELEKTRO
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember
Surabaya 2017



TUGAS AKHIR - TE 141599

**RANCANG BANGUN *BUOYANT BOAT* BERTENAGA
SURYA UNTUK MEMBANTU KOMUNIKASI ANTARA
REMOTELY OPERATED VEHICLE UNDERWATER
ROBOT DENGAN *GROUND STATION***

Badrut Tamam
NRP 07111440000101

Dosen Pembimbing
Ronny Mardiyanto, ST., MT., Ph.D.

DEPARTEMEN TEKNIK ELEKTRO
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember
Surabaya 2017



FINAL PROJECT - TE 141599

***DESIGN AND DEVELOPMENT OF BUOYANT BOAT
SOLAR POWERED TO ASSIST COMMUNICATION
BETWEEN REMOTELY OPERATED VEHICLE
UNDERWATER ROBOT AND GROUND STATION***

Badrut Tamam
NRP 07111440000101

Supervisor
Ronny Mardiyanto, ST., MT., Ph.D.

ELECTRICAL ENGINEERING DEPARTMENT
Faculty of Electrical Technology
Sepuluh Nopember Institute of Technology
Surabaya 2017

PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul “RANCANG BANGUN *BUOYANT BOAT* BERTENAGA SURYA UNTUK MEMBANTU KOMUNIKASI ANTARA *REMOTELY OPERATED VEHICLE UNDERWATER ROBOT* DENGAN *GROUND STATION*” adalah benar-benar hasil karya intelektual sendiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya orang lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai dengan peraturan yang berlaku.

Surabaya, 25 Desember 2017

Badrut Tamam
NRP. 07111440000101

.....*Halaman ini sengaja dikosongkan*.....

**RANCANG BANGUN BUOYANT BOAT BERTENAGA
SURYA UNTUK MEMBANTU KOMUNIKASI
ANTARA REMOTELY OPERATED VEHICLE
UNDERWATER ROBOT DENGAN GROUND STATION**

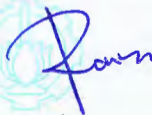
TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik
Pada**

**Bidang Studi Elektronika
Departemen Teknik Elektro
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember**

Menyetujui :

Dosen Pembimbing



Ronny Mardiyanto, ST., MT., Ph.D.
NIP. 198101182003121003

No. Agenda	Tgl. Terima	PERPUSTAKAAN ITS
Terima Dari		
	H/10/2017	



.....*Halaman ini sengaja dikosongkan*.....

Rancang Bangun *Buoyant Boat* Bertenaga Surya untuk Membantu Komunikasi antara *Remotely Operated Vehicle Underwater Robot* dengan *Ground Station*

Nama : Badrut Tamam
Pembimbing : Ronny Mardiyanto, ST., MT., Ph.D.

ABSTRAK

Buoyant Boat merupakan wahana apung untuk membantu komunikasi antara ROV (*Remotely Operated Vehicle*) *underwater robot* dengan *ground station*. Komunikasi antara *Buoyant Boat* dengan *ground station* secara *wireless* meliputi kendali dan data, sedangkan dengan ROV melalui kabel berupa data, kendali, dan suplai daya.

Komunikasi antara ROV dengan *ground station* ada dua cara, yaitu secara *wireless* dan melalui kabel. Namun, komunikasi *wireless* tidak pernah digunakan untuk observasi di perairan dalam. Hal ini karena terdapat redaman sinyal yang besar pada perbatasan air dengan udara. Oleh karena itu, dirancanglah *Buoyant Boat* yang menggunakan metode sistem komunikasi *buoyant*, dimana *Buoyant Boat* dapat membantu komunikasi dan suplai daya untuk ROV yang akan melakukan observasi dibawah air. Kendali menggunakan sepasang *remote control* dengan *receiver* dan komunikasi data menggunakan telemetri. *Buoyant Boat* memiliki tiga mode kendali, yakni sistem kendali manual, *auto*, dan *loiter* (atau *lock location*). Kebutuhan daya disuplai oleh tiga buah panel surya yang terpasang pada *Buoyant Boat*.

Dari pengujian, didapatkan bahwa kontrol manual *Buoyant Boat* bekerja dengan baik. Pada mode *auto*, *Buoyant Boat* dapat melalui 3 titik koordinat yang ditentukan dengan kecepatan maksimal 1.37 Kmph. Dalam mode *loiter*, *Buoyant Boat* dapat kembali ke titik *lock location* ketika *Buoyant Boat* terbawa arus sejauh radius 2 meter. Tiga panel surya yang digunakan dapat menyuplai daya mulai pukul 06.09 WIB hingga 16.34 WIB. Panel surya yang digunakan menghasilkan daya total 82,8 W dengan efisiensi sebesar 27,6%. Realisasi *Buoyant Boat* memiliki dimensi 255 x 110 x 38 cm.

Kata kunci: *Buoyant Boat*, Daya, Kendali, Komunikasi.

.....*Halaman ini sengaja dikosongkan*.....

Design and Development of Buoyant Boat Solar Powered to Assist Communication between Remotely Operated Vehicle Underwater Robot and Ground Station

Name : Badrut Tamam

Supervisor : Ronny Mardiyanto, ST., MT., Ph.D.

ABSTRACT

Buoyant Boat is a floating vehicle to assist communication between ROV (Remotely Operated Vehicle) underwater robot with ground station. Buoyant boat with ground station is communicated by wireless which include control and data, while Buoyant Boat with ROV is connected by cable which consist control, data, and power supply.

There are two ways communication between ROV and ground station, by wireless and through a wire. Nevertheless, wireless communication system never used to observed on the deep waters because there are high attenuation of a signal when cross the air-water boundary. Base on these problem, Buoyant Boat that used buoyant communication system has been developed to assist the communication and to supplai a power to ROV which will observe underwater. Control in this robot used a pair of remote control and receiver, while data communication is used a telemetri. Buoyant Boat have three modes control, that are manual, auto, and loiter (or lock location). Power needed is suplied by three photovoltaic that was installed on Buoyant Boat.

Based on experiment, we know that manual control can work properly. In auto mode, Buoyant Boat can going through 3 point that have had set on the code program before with maximum speed 1.37 Kmph. On loiter mode, Buoyant Boat can back to the lock location when Buoyant Boat has moved as far as 2 meters from lock location. Three PV that was used, can powered from 06.09 AM until 04.34 PM. The three photovoltaic provided 82,8 W. It mean that photovoltaic has 27,6% of eficiency. Buoyant Boat have dimension length x width x high: 255 x 110 x 38 cm.

Keywords: Buoyant Boat, Communication, Control, Power.

.....*Halaman ini sengaja dikosongkan*.....

KATA PENGANTAR

Alhamdulillah, syukur yang tiada henti penulis panjatkan kehadirat Allah SWT serta tidak lupa sholawat serta salam semoga tetap tercurah kepada junjungan kita Nabi Muhammad SAW sehingga penelitian dalam tugas akhir ini bisa berjalan lancar dan selesai tepat pada waktunya.

Selama pelaksanaan penelitian Tugas Akhir ini, penulis mendapatkan bantuan dari berbagai pihak, dan penulis sampaikan rasa terima kasih. Terima kasih yang sebesar-besarnya juga penulis sampaikan kepada berbagai pihak yang mendukung dan membantu dalam tugas akhir ini, diantaranya:

1. Kedua orang tua yang senantiasa memberikan do'a, nasihat, motivasi, dukungan dan karena keberadaan merekalah penulis tetap semangat untuk menyelesaikan penelitian ini.
2. Ronny Mardiyanto, ST., MT., Ph.D. selaku dosen pembimbing, atas dukungan moriil maupun materiil, serta bimbingan, inspirasi, pengarahan, dan motivasi yang diberikan selama pengerjaan penelitian tugas akhir ini.
3. Seluruh dosen-dosen Departemen Teknik Elektro, khususnya dosen-dosen bidang studi Elektronika.
4. Rekan-rekan yang banyak membantu dalam penyelesaian tugas akhir ini.

Penulis sadar bahwa Tugas Akhir ini masih belum sempurna dan masih banyak hal yang perlu diperbaiki. Saran, kritik dan masukan baik dari semua pihak sangat membantu penulis terutama untuk berbagai kemungkinan pengembangan lebih lanjut.

Terakhir, penulis berharap Tugas Akhir ini dapat memberikan manfaat bagi banyak pihak dan dapat membantu pengembangan tentang ROV *underwater robot* kedepannya.

Surabaya, 25 Desember 2017

Badrut Tamam

.....*Halaman ini sengaja dikosongkan*.....

DAFTAR ISI

PERNYATAAN KEASLIAN	i
TUGAS AKHIR	iii
ABSTRAK	v
ABSTRACT	vii
KATA PENGANTAR	ix
DAFTAR ISI	xi
DAFTAR GAMBAR	xv
DAFTAR TABEL	xvii
BAB I. PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Perumusan Masalah	3
1.3. Tujuan Penelitian	3
1.4. Batasan Masalah	3
1.5. Metodologi Penelitian	3
1.6. Sistematika Penulisan	6
1.7. Relevansi	6
BAB II. TINJAUAN PUSTAKA	7
2.1. Robot Bawah Air (<i>Underwater Robot</i>)	7
2.2. ROV (<i>Remotely Operated Vehicle</i>)	7
2.3. Komunikasi Bawah Air	8
2.4. <i>Acoustic Underwater Positioning</i>	10
2.5. Arduino Mega 2560	11
2.5.1. Sumber Daya/ <i>Power</i>	13
2.5.2. Memori	13
2.5.3. <i>Input/Output</i>	13
2.5.4. Komunikasi	14
2.5.5. Perlindungan Arus USB	15
2.6. PWM (<i>Pulse Wide Modulation</i>)	15
2.7. PID (<i>Proportional – Integral – Derivative</i>)	16
2.8. <i>Driver Motor</i>	17
2.9. <i>Bilge Pump Motor</i>	18
2.10. RC (<i>Radio Control</i>)	19
2.11. Telemetri	20
2.12. <i>Global Positioning System</i> (GPS)	22
2.13. Kompas	23
2.14. <i>Photovoltaic</i> (PV)	23

2.15. MPPT <i>Solar Charge Controller</i>	24
2.16. <i>Adjustable Buck Converter</i>	25
2.17. Baterai Lipo (<i>Litium Polimer</i>).....	26
2.18. AutoCAD 2015 (<i>Student Version</i>)	27
2.19. Tinjauan Pustaka	28
2.19.1. <i>Underwater Communication Systems: A Review</i> [9] ..	28
2.19.2. <i>Solar Powered Ferry Boat for the Rural Area of Bangladesh</i> [37].....	28
2.19.3. <i>Autonomous Surface Vehicle for Measuring Water Body Parameters</i> [38].....	28
2.19.4. <i>Evolutionary Path Planning of an Autonomous Surface Vehicle for Water Quality Monitoring</i> [39]...29	
2.19.5. <i>Design of Small Autonomous Boat for Course-keeping Manuvers</i> [40]	29
2.19.6. <i>Development of Unmanned Surface Vehicle for Smart Water Quality Inspector</i> [41]	29
2.19.7. <i>Green Sailing of Solar PV Powered Country Boat using Buck-Boost Chopper</i> [42].....	30
2.19.8. <i>Research Solar Power Boat – Data Management and Online Visualization</i> [43]	30
2.19.9. <i>Soft-Switching PWM Forward Power Converter Auxiliary Active Clamped Capacitor for Solar Energy-Driven Boat</i> [44].....	30
BAB III. PERANCANGAN SISTEM	31
3.1. Diagram Blok Sistem	33
3.2. Perancangan Perangkat Keras	35
3.2.1. Perahu Karet Sebagai Kapal.....	35
3.2.2. Tata Letak Panel Surya.....	35
3.2.3. Posisi Motor	37
3.2.4. Desain <i>Propeller</i>	38
3.2.5. Tata Letak Komponen Elektronik Utama.....	38
3.2.6. Tata Letak Komponen Elektronik Komunikasi.....	39
3.2.7. Panel Surya	40
3.2.8. MPPT <i>Solar Charge Controller</i>	40
3.2.9. <i>Buck-Converter</i>	41
3.2.10. Arduino Mega 2560	41
3.2.11. GPS	43
3.2.12. Kompas	44
3.2.13. <i>Remote Control dan Receiver</i>	45

3.2.14. Telemetry.....	46
3.2.15. <i>Driver Motor</i>	47
3.2.16. <i>Bilge Pump Motor</i>	47
3.2.17. <i>Analog to Digital Converter</i>	47
3.3. Perancangan Perangkat Lunak.....	48
3.3.1. Perancangan <i>Main Program</i> (Lampiran A)	48
3.3.2. Perancangan Kendali Manual (Lampiran B)	50
3.3.3. Perancangan Kendali <i>Auto</i> (Lampiran C).....	52
3.3.4. Perancangan Mode <i>Loiter</i> (Lampiran D).....	54
3.3.5. Perancangan Kontrol PID.....	55
3.3.6. Perhitungan Kebutuhan Daya.....	56
3.3.7. Pembacaan ADC (<i>Analog to Digital Converter</i>)	57
3.3.8. Antarmuka Data Pada Monitor dengan PLX-DAQ v2.10 (Lampiran E).....	57
3.3.9. Antarmuka Data Pada Monitor.....	58
BAB IV. PENGUJIAN DAN ANALISIS	61
4.1. Pengujian GPS.....	62
4.2. Pengujian Kompas Digital.....	63
4.3. Pengujian Kendali	64
4.3.1. Pengujian Kendali Manual	65
4.3.2. Pengujian Mode <i>Auto</i>	67
4.3.3. Pengujian Mode <i>Loiter</i>	70
4.4. Pengujian Panel Surya	72
4.5. Monitoring Data	73
4.5.1. PLX-DAQ v2.10	73
4.5.2. <i>Graphical User Interface</i> (GUI).....	75
4.6. Analisis Keseluruhan Perancangan Sistem	75
BAB V. PENUTUP	77
5.1. Kesimpulan.....	77
5.2. Saran.....	77
DAFTAR PUSTAKA	79
LAMPIRAN A	85
LAMPIRAN B	87
LAMPIRAN C	89
LAMPIRAN D	93
LAMPIRAN E	97
LAMPIRAN F	99
LAMPIRAN G	101
BIODATA PENULIS	103

.....*Halaman ini sengaja dikosongkan*.....

DAFTAR GAMBAR

Gambar 1.1 ROV Pencari <i>Black Box</i> AirAsia QZ8501 [3].	1
Gambar 2.1 Venom 3k-200 <i>Workclass</i> ROV [8].	8
Gambar 2.2 Arsitektur Sistem Komunikasi Buoyant [9].	9
Gambar 2.3 Arsitektur Sistem Komunikasi Langsung [9].	10
Gambar 2.4 (a) Skema Sistem USBL; (b) Skema Sistem LBL [10].	11
Gambar 2.5 <i>Board</i> Arduino Mega 2560 [11].	12
Gambar 2.6 <i>Duty Cycle</i> pada PWM [12].	15
Gambar 2.7 Skema PID Kontrol [15].	16
Gambar 2.8 <i>Driver Motor</i> VN2SP30 [17].	17
Gambar 2.9 Skematik Rangkaian VN2SP30 [18].	18
Gambar 2.10 <i>Bilge Pump Motor</i> [20].	18
Gambar 2.11 Perlengkapan RC (a) DX7s RC <i>Transmitter</i> ; (b) <i>Orange RX Receiver</i> [21].	20
Gambar 2.12 <i>Telemetry Kit</i> 433 MHz [23].	21
Gambar 2.13 U-Blox Neo M8N <i>with Compass</i> [26].	22
Gambar 2.14 Kompas Digital HMC5883L [29].	23
Gambar 2.15 <i>Photovoltaic</i> (Solar Panel) 100Wp.	24
Gambar 2.16 Konsep <i>Charge Controller</i> [31].	24
Gambar 2.17 MPPT <i>Solar Charge Controller</i> T40.	25
Gambar 2.18 <i>Adjustable Buck Converter</i> [33].	25
Gambar 2.19 Baterai Lipo 3 Sel.	27
Gambar 2.20 Logo AutoCAD [35].	27
Gambar 3.1. Skema Sistem Keseluruhan.	31
Gambar 3.2. Diagram Blok Sistem.	34
Gambar 3.3. Perahu Karet [45].	35
Gambar 3.4. Perancangan Tempat Panel Surya.	36
Gambar 3.5. Pemasangan Tiga Modul Panel Surya Pada Buoyant Boat.	36
Gambar 3.6. Desain Tempat (Dudukan) Motor.	37
Gambar 3.7. Posisi Motor pada Kapal.	37
Gambar 3.8. <i>Propeller</i> .	38
Gambar 3.9. Komponen Elektronik Utama.	38
Gambar 3.10. Desain Peletakan Komponen Elektronik Komunikasi.	39
Gambar 3.11. Terminal Penghubung.	40
Gambar 3.12. Konfigurasi pada MPPT <i>Solar Charge Controller</i> .	40
Gambar 3.13. XL4005 <i>Typical Application Circuit</i> (24V~5V/5A) [32].	41

Gambar 3.14. Skematik Arduino Mega 2560 [46].	42
Gambar 3.15. Skematik GPS U-Blox M8N [47].	43
Gambar 3.16. Antarmuka GPS dengan Arduino Mega 2560	44
Gambar 3.17. Skematik Kompas HMC5883L [48].	44
Gambar 3.18. Antarmuka Kompas HMC5883L dengan Arduino Mega 2560.	45
Gambar 3.19. Tombol-tombol Remot Kontrol untuk Kendali Buoyant Boat.	45
Gambar 3.20. Antarmuka <i>Receiver</i> R9D dengan Arduino Mega 2560.	46
Gambar 3.21. Antarmuka Telemetri dengan Arduino Mega 2560.	46
Gambar 3.22. Antarmuka <i>Driver Motor</i> VNH2SP30 dengan Arduino Mega 2560.	47
Gambar 3.23. <i>Bilge Pump Motor</i> pada <i>Buoyant Boat</i> .	47
Gambar 3.24. Rangkain Pembagi Tegangan.	48
Gambar 3.25. Diagram Alir <i>Switching Mode</i> .	49
Gambar 3.26. Diagram Alir Kendali Manual.	50
Gambar 3.27. Diagram Alir Mode <i>Auto</i> .	52
Gambar 3.28. Diagram Alir Mode <i>Loiter</i> .	54
Gambar 3.29. Diagram Blok Kontrol PID.	55
Gambar 3.30. PLX-DAQ v2.10.	58
Gambar 3.31. Antarmuka Monitoring Data.	59
Gambar 4.1. <i>Buoyant Boat</i> Tampak Belakang.	61
Gambar 4.2. Danau 8 ITS.	62
Gambar 4.3. Pemetaan Sudut GPS.	63
Gambar 4.4. Pemetaan Sudut Kompas.	64
Gambar 4.5. Pengujian Kendali Manual di Kolam Teknik Fisika ITS.	65
Gambar 4.6. Pengujian Kendali Manual di Danau 8 ITS dengan 1 PV.	66
Gambar 4.7. Pengujian Kendali Manual di Danau 8 ITS dengan 3 PV.	66
Gambar 4.8. Plot Titik Tujuan <i>Buoyant Boat</i> .	67
Gambar 4.9. <i>Trajectory Buoyant Boat Satellite View</i> .	68
Gambar 4.10. <i>Trajectory Buoyant Boat Maps View</i> .	68
Gambar 4.11. Pengujian Mode <i>Loiter</i> .	71
Gambar 4.12. Pemetaan Nilai <i>Error</i> Sudut.	71
Gambar 4.13. Pengujian 3 Modul Panel Surya.	72
Gambar 4.14. Grafik Arus Terhadap Waktu.	73
Gambar 4.15. Tampilan PLX-DAQ v2.10 saat <i>Monitoring Data</i> .	74
Gambar 4.16. GUI untuk <i>Monitoring Data</i> .	75

DAFTAR TABEL

Tabel 2.1. Spesifikasi Teknik Arduino Mega 2560 [11].	12
Tabel 2.2. Spesifikasi <i>Driver Motor</i> VNH2SP30 [16].	17
Tabel 2.3. Spesifikasi <i>Bilge Pump Motor</i> [19].	19
Tabel 2.4. Spesifikasi Telemetry [24].	21
Tabel 2.5. Spesifikasi U-Blox Neo-M8N GPS <i>Compass</i> [27].	22
Tabel 2.6. Kelebihan dan Kekurangan Baterai Lipo [34].	26
Tabel 3.1. Spesifikasi <i>Buoyant Boat</i> .	32
Tabel 4.1. Data GPS.	62
Tabel 4.2. Data Sudut Kompas.	63
Tabel 4.3. Data <i>Trajectory Buoyant Boat</i> .	69
Tabel 4.4. Pengujian Fitur <i>Overall</i> .	76

.....*Halaman ini sengaja dikosongkan*.....

BAB I

PENDAHULUAN

1.1. Latar Belakang

Luas wilayah perairan Indonesia yang diakui United Nation Convention of The Sea (UNCLOS) mencapai 2/3 dari luas wilayah Indonesia. Wilayah yang sangat luas tersebut merupakan suatu peluang sehingga kegiatan riset geologi dan geofisika mulai tertuju ke laut. Sebelum adanya ROV (*Remotely Operated Vehicle*), observasi bawah laut dilakukan oleh para penyelam. Selanjutnya, ROV mulai dikembangkan karena manusia memiliki keterbatasan seperti tidak mampu menahan tekanan yang besar, suhu ekstrim, dan lain-lain [1].

ROV, wahana selam mini, berfungsi untuk membantu manusia dalam kegiatan eksplorasi objek bawah laut terutama yang tidak bisa dijangkau oleh manusia seperti objek dalam gua kecil di bawah laut dengan kedalaman 3000 meter [2]. Saat kecelakaan pesawat AirAsia QZ8501, ROV berperan dalam pencarian *Black Box* di bawah laut. Gambar 1.1 menunjukkan ROV yang digunakan untuk mencari *Black Box* AirAsia tersebut [3].



Gambar 1.1 ROV Pencari *Black Box* AirAsia QZ8501 [3].

Teknologi ROV bahkan sudah menjadi tumpuan utama operasi bawah laut pada industri minyak dan gas lepas pantai selama satu dekade terakhir. Serta, memiliki peran penting dalam energi terbarukan laut (*Marine Renewable Energy*) dan industri rekayasa bangunan laut untuk keperluan militer, keperluan observasi laut, dan lainnya. Kegunaan ROV secara spesifik misalnya untuk pemeriksaan, perbaikan, dan perawatan pipa gas bawah laut. Selain itu, masih banyak lagi aplikasi dari ROV yang sangat bermanfaat [4].

Umumnya, ROV terhubung dengan kabel secara langsung dengan pilot di pulau atau di atas sebuah kapal [5]. Sistem ROV terpasang pada kapal tersebut. Kapal dilengkapi dengan DGPS (*Differential Global Positioning System*) sebagai penentuan posisi utama. Sedangkan, untuk penentuan posisi ROV di bawah laut menggunakan gelombang suara atau *Acoustic Underwater Positioning* [6]. Pilot diam ditempat saat ROV bergerak menyusuri bawah laut, sehingga area gerak ROV bergantung pada panjang kabel. Hal ini menyebabkan pergerakan ROV menjadi terbatas [3].

Pada praktiknya, komunikasi antara ROV dengan *ground station* ada dua cara, yaitu selain melalui kabel juga dengan cara *wireless*. Namun, komunikasi secara *wireless* tidak pernah digunakan untuk observasi di laut atau perairan dalam. Hal tersebut terjadi karena terdapat redaman sinyal pada perbatasan air dengan udara yang disebabkan oleh fenomena pematulan ataupun pembiasan [5] [7].

Berdasarkan permasalahan diatas, dirancanglah *Buoyant Boat Bertenaga Surya* untuk Membantu Komunikasi antara ROV *Underwater Robot* dengan *Ground Station*. *Buoyant Boat* ini posisinya dapat sejajar dengan ROV. *Buoyant Boat* dengan ROV terhubung melalui kabel, sedangkan dengan *ground station* terhubung secara *wireless*. Selain itu, juga dilengkapi dengan GPS untuk menentukan posisi kapal utama dan ROV. Kontrol pada *Buoyant Boat* ini ada tiga mode, yaitu mode manual, *auto*, dan *loiter*. Dimana mode *loiter* ini digunakan untuk membuat posisi *Buoyant Boat* tetap pada lokasi yang dikunci (*lock location*). Tidak hanya itu, *Buoyant Boat* menggunakan panel surya untuk memenuhi kebutuhan daya ROV dan *Buoyant Boat* itu sendiri. Dengan *Buoyant Boat* ini diharapkan komunikasi antara ROV dengan *ground station* dapat dilakukan dari jarak jauh, sehingga jarak jangkauan ROV menjadi lebih luas.

1.2. Perumusan Masalah

Permasalahan yang dibahas dalam tugas akhir ini adalah:

1. Komunikasi antara ROV dengan *ground station*.
2. *Self charging* pada *Buoyant Boat* untuk membantu komunikasi antara ROV dengan *ground station*.
3. Kontrol posisi *Buoyant Boat* dengan menggunakan GPS.

1.3. Tujuan Penelitian

Penelitian pada tugas akhir ini bertujuan sebagai berikut:

1. Membantu komunikasi antara ROV dengan *ground station*.
2. Mengatur sistem kelistrikan untuk pengisian baterai *Buoyant Boat* yang bersumber dari PV.
3. Membuat sistem kontrol posisi *Buoyant Boat* agar posisinya dapat *lock location*.

1.4. Batasan Masalah

Batasan masalah dari tugas akhir ini adalah sebagai berikut:

1. Sistem ini hanya diaplikasikan untuk kondisi cuaca cerah, cerah berawan, atau ada cahaya matahari.
2. *Photovoltaic* dalam sistem digunakan secara langsung, tanpa ada pengembangan untuk meningkatkan efisiensi.
3. Tidak ikut serta dalam pembuatan ROV.
4. Desain kapal menggunakan perahu karet.

1.5. Metodologi Penelitian

Langkah-langkah yang dikerjakan pada tugas akhir ini adalah sebagai berikut:

1. Studi literatur

Studi literatur berisi serangkaian kegiatan pengumpulan dan pengkajian dasar teori yang relevan dan terpercaya untuk menunjang penulisan tugas akhir ini. Literatur dapat bersumber dari paper, jurnal, artikel, buku, maupun *website*, yang bertaraf nasional dan internasional, serta dari hasil konsultasi dengan dosen pembimbing. Namun, terdapat satu sumber rujukan utama yang telah dilakukan yakni *IEEE Explore*.

2. Observasi dan Analisa Masalah

Pada tahap ini dilakukan pengkajian terhadap sistem-sistem yang ada pada ROV *underwater robot*. Analisa terhadap kekurangan-kekurangan ROV bertujuan untuk melakukan penyempurnakan terhadap ROV tersebut. Observasi dan analisa masalah dilakukan dengan mengkaji paper, jurnal, dan isu-isu terkini di *website* terkait ROV. Salah satu masalah yang ditemukan saat pengkajian berbagai sumber adalah komunikasi antara ROV dengan *ground station* jika menggunakan komunikasi secara *wireless*. Selanjutnya melakukan inovasi yang implementatif untuk mengatasi masalah tersebut.

3. Persiapan Alat dan Bahan

Persiapan dilakukan agar dalam perancangan alat menjadi lebih terstruktur. Tahap ini merupakan tahap pencarian informasi mengenai konsep-konsep yang dibutuhkan untuk merancang *Buoyant Boat* ini, yang bisa didapatkan dari studi literatur disertai dengan bimbingan yang intensif dengan dosen. Kemudian dilakukan pengumpulan alat dan bahan yang dibutuhkan. Tempat pembelian alat dan bahan yaitu toko Berkat Elektronik, Isee, Digiware Store, Pasar Genteng, dan pilihan terakhir adalah toko *online*. Alat dan bahan yang diperlukan adalah seperangkat solder, komputer, *cutter*, gunting, bodi kapal, GPS, PV, kabel *umbilical*, *charge controller*, *radio control*, motor, serta alat dan bahan pendukung lainnya.

4. Perancangan *Buoyant Boat*

Perancangan bertujuan mendapatkan desain dan mekanisme yang optimal dengan memperhatikan data-data sebelumnya. Perancangan disini termasuk perancangan *software* dan *hardware*. *Software* dirancang dengan pembuatan *source code* yang meliputi sistem kendali *Buoyant Boat*, kontrol posisi, sistem *charge controller*, dan sistem komunikasi data antara *ground station* dengan *Buoyant Boat* dan antara *Buoyant Boat* dengan ROV. Konsep utama dari *Buoyant Boat* ini yaitu dapat bergerak bebas di lautan atau perairan dalam waktu yang relatif lama. Dalam hal ini, peran PV sangat vital karena menjadi penyuplai daya utama. Desain *hardware* meliputi bodi kapal, arsitektur elektronik, *interface* dan desain mekanik penunjang lainnya, seperti tata letak PV dan motor. Aktifitas perancangan dilakukan di laboratorium Mikrokontroler (A206), Departemen Teknik Elektro dan tempat tinggal.

5. Pembuatan *Buoyant Boat*

Proses pembuatan dilakukan atas dasar perencanaan yang matang. Tahap ini adalah tahap pengintegrasian antara *software* dengan *hardware*. Program yang dibuat diimplementasikan pada masing-masing modul atau bagian sistem. Pembuatan mekanik serta tata letaknya direalisasikan pada tahap ini. Akhir tahap ini adalah penyatuan semua bagian menjadi sebuah *Buoyant Boat* utuh, namun tanpa ROV.

6. Tahap Pengujian

Pengujian dilakukan bertahap. Pertama dilakukan pengujian pada masing-masing bagian agar pemecahan masalah lebih mudah dilakukan. *Software* diuji terlebih dahulu sebelum diimplementasikan ke *hardware*. Kemudian, dilanjutkan dengan pengujian *hardware* yang sudah tertanam *software* yang telah dibuat. Bagian-bagian tersebut meliputi sistem komunikasi data antara *Buoyant Boat* dengan *ground station* dan sistem kendali pada *Buoyant Boat*, pengujian *self charging* dari PV, dan pengujian kontrol posisi *Buoyant Boat*. Pengujian kedua yaitu dengan menguji kinerja *Buoyant Boat* secara keseluruhan.

Tahap pengujian dilakukan untuk menentukan keandalan dari sistem yang telah dibuat, *software* maupun *hardware*. Lokasi pengujian *software* dilakukan di laboratorium Mikrokontroler A206 dengan arahan dari dosen pembimbing. Sedangkan, pengujian *hardware* dan pengujian total dilakukan di danau delapan ITS.

7. Analisa dan Evaluasi

Analisa dilakukan terhadap hasil pengujian sehingga karakteristik *software* dan *hardware* dapat diketahui. Analisa dilakukan pada tiga bagian utama, yaitu komunikasi antara *Buoyant Boat* dengan *ground station*, sistem *self charging* dari PV, dan sistem kendali serta kontrol posisi *Buoyant Boat* agar selalu sejajar dengan ROV. Apabila karakteristik dari bagian-bagian tersebut belum sesuai, maka perlu dilakukan evaluasi pada sistem untuk dirancang dan diuji kembali.

8. Penyusunan Laporan Tugas Akhir

Tahap penyusunan laporan merupakan tahap terakhir dari proses pengerjaan tugas akhir ini. Laporan berisi seluruh hal yang berkaitan dengan tugas akhir yang telah dikerjakan yaitu meliputi pendahuluan, studi literatur, tinjauan pustaka, perancangan dan pembuatan sistem, pengujian dan analisa, serta penutup.

1.6. Sistematika Penulisan

Dalam buku tugas akhir ini, pembahasan mengenai sistem yang dibuat terbagi menjadi Lima Bab dengan sistematika penulisan sebagai berikut:

- Bab I : Pendahuluan
Bab ini meliputi penjelasan latar belakang, rumusan masalah, batasan masalah, tujuan, metodologi, sistematika penulisan, dan relevansi.
- Bab II : Tinjauan Pustaka
Bab ini menjelaskan tentang teori penunjang yang terkait maupun yang dibutuhkan dalam pengerjaan tugas akhir ini. Dasar teori yang menunjang meliputi Robot Bawah Air, ROV, Komunikasi Bawah Air, *Acoustic Underwater Positioning*, *Radio Control*, PV, Arduino Mega, Arduino IDE, GPS, *Solar Charge Controller*, Kompas, *Bilge Pump Motor*, *Driver Motor*, Lipo, *DC Adjustable Buck Converter*, *3D Printer*, AutoCAD 2015 (*Student Version*). Selain itu, bagian ini juga berisi tinjauan pustaka untuk membandingkan sistem yang telah ada dengan yang dirancang pada tugas akhir ini.
- Bab III : Perancangan Sistem
Bab ini menjelaskan tentang perencanaan sistem yang meliputi perangkat keras (*hardware*) dan perangkat lunak (*software*) untuk pembuatan *Buoyant Boat*.
- Bab IV : Pengujian dan Analisis
Bab ini menjelaskan tentang hasil uji coba alat beserta analisisnya.
- Bab V : Penutup
Bab ini berisi tentang kesimpulan yang diperoleh dari pembuatan alat serta saran untuk pengembangan lebih lanjut.

1.7. Relevansi

Buoyant Boat merupakan suatu inovasi baru untuk membantu komunikasi antara *ground station* dengan ROV secara *wireless* tanpa gangguan pada sinyal yang ditransmisikan. Dengan *Buoyant Boat*, operator ROV tidak perlu terjun ke lokasi observasi secara langsung. Cukup dengan mengoperasikan *Buoyant Boat* dari tempat yang nyaman dan aman, dan *Buoyant Boat* lah nantinya yang membantu komunikasi jarak jauh dengan *ground station*.

BAB II

TINJAUAN PUSTAKA

2.1. Robot Bawah Air (*Underwater Robot*)

Robot bawah air adalah salah satu jenis robot tanpa awak yang pengaplikasiannya untuk kegiatan observasi bawah air. Jenis robot bawah air ada dua, yaitu *Autonomous Underwater Vehicles* (AUVs) dan *Remotely Operated Vehicles* (ROVs). AUVs adalah robot bawah air yang dapat beroperasi sendiri sesuai perintah pada program yang ditanamkan didalam chip yang digunakan robot AUV tersebut. Seperti namanya, “*autonomous*”, robot ini bersifat *auto*. Biasanya, robot ditunjang dengan kamera, lampu sebagai pencahayaan, dan sensor-sensor pendukung dalam melakukan operasi dibawah laut. Sedangkan, ROVs sesuai namanya “*remotely*”, berarti ada pengendali atau operator yang mengendalikan kerja dari robot ROV dengan menggunakan perangkat kendali (*remote control*). Sama dengan AUVs, ROVs umumnya juga dilengkapi dengan kamera, lampu, dan sensor-sensor pendukung lainnya. Hanya saja, kamera pada ROV selain untuk merekam atau mengambil gambar objek juga sebagai penunjuk arah bagi operator [6].

2.2. ROV (*Remotely Operated Vehicle*)

Seperti penjelasan sebelumnya, ROV bergerak dibawah kendali operator yang terhubung melalui kabel *umbilical*. Melalui kabel *umbilical* ini energi listrik, data video, dan sinyal dikirimkan ke operator ROV untuk diamati. Peran ROV dibawah laut meliputi bidang militer, industri, observasi, dan bidang-bidang lainnya. Dalam bidang militer, angkatan laut Amerika Serikat merupakan yang pertama memanfaatkannya pada tahun 1960-an. Yakni, mereka memanfaatkan ROV yang secara khusus bertujuan mengambil ranjau-ranjau di dasar laut dan pada saat peristiwa hilangnya bom atom saat terjadi kecelakaan pesawat di Spanyol. Dari bidang industri, industri tambang minyak dan gas yang melakukan eksplorasi di laut dalam menjadi yang terdepan dalam memanfaatkan ROV. Bahkan hal tersebut telah menjadi tren selama sepuluh tahun terakhir [4].

Secara umum, kamera dan lampu pencahayaan menjadi kelengkapan tambahan yang utama. Namun, untuk mendukung kemampuannya, ROV kemudian dilengkapi dengan sensor-sensor

lain, tergantung pada tujuan pembuatan ROV itu sendiri. Peralatan atau sensor yang biasa digunakan yaitu kamera, lampu, transponder, kompas, odometer, *bathy* (data kedalaman), sonar, magnetometer, lengan robotik, pengukur suhu, pengambil sampel air, dan lain sebagainya [3].

Jenis-jenis ROV berdasarkan ukuran, berat, dan kekuatannya dibedakan menjadi lima, yaitu *mikro ROV*, *mini ROV*, *light workclass*, *heavy workclass*, dan *trenching/burial*. Tipe mikro dan mini ROV digunakan untuk kedalaman kurang dari 300 m. Pemanfaatannya untuk inspeksi dan pengamatan, misal untuk inspeksi waduk, inspeksi perairan pantai, inspeksi saluran air, dan inspeksi nuklir. Tipe *workclass* banyak digunakan untuk pengeboran lepas pantai, survei dan rektifikasi pipa gas bawah laut. Contoh tipe *workclass* adalah Venom yang ditunjukkan pada gambar 2.1. Venom dapat beroperasi hingga kedalaman 3000 m dan digerakkan menggunakan tenaga hidrolik agar daya yang dihasilkan sebesar 150 HP [6].



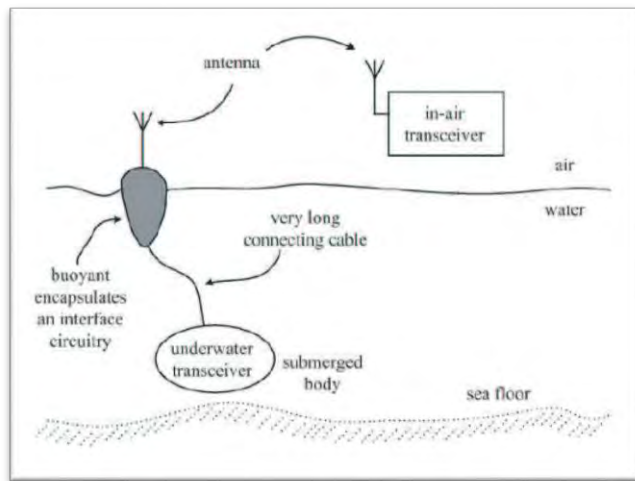
Gambar 2.1 Venom 3k-200 *Workclass* ROV [8].

2.3. Komunikasi Bawah Air

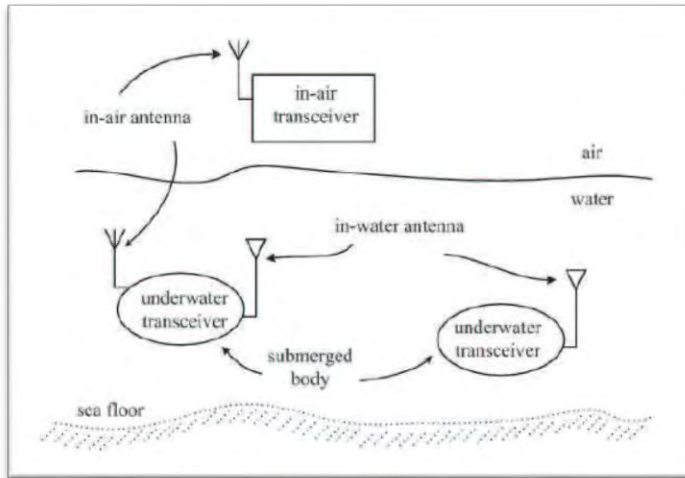
Komunikasi bawah air dapat menggunakan gelombang sonar atau gelombang elektromagnetik. Sistem komunikasi sonar ada dua yaitu sonar pasif dan sonar aktif. Sistem komunikasi sonar pasif mengambil gelombang sonar yang dipropagasikan dibawah air. Sistem komunikasi sonar aktif menerima gelombang sonar dengan baik seperti saat mengemisikan gelombang sonar dibawah air. Keduanya menggunakan *hydrophones* sebagai pemancar dan

penerima gelombang. Sistem komunikasi dengan gelombang sonar tidak dapat digunakan pada sistem komunikasi yang melalui air dan udara karena besarnya pelemahan amplitudo pada gelombang sonar ketika melalui air dan udara.

Sistem komunikasi elektromagnetik merupakan alternatif untuk komunikasi yang melalui air dan udara karena hampir tidak mengurangi kecepatan dan pelemahan amplitudo. Sistem ini menggunakan antenna untuk memancarkan dan menerima gelombang elektromagnetik. Terdapat dua arsitektur dengan sistem ini yaitu sistem buoyant dan sistem hubungan langsung. Pada sistem buoyant, gelombang hanya melalui satu pasang antenna yang ada di udara, sedangkan sistem hubungan langsung menggunakan antenna yang dipasang di udara dan satu pasang antenna di dalam air. Arsitektur sistem buoyant ditunjukkan pada gambar 2.2 dan sistem hubungan langsung ditunjukkan oleh gambar 2.3 [9].



Gambar 2.2 Arsitektur Sistem Komunikasi Buoyant [9].



Gambar 2.3 Arsitektur Sistem Komunikasi Langsung [9].

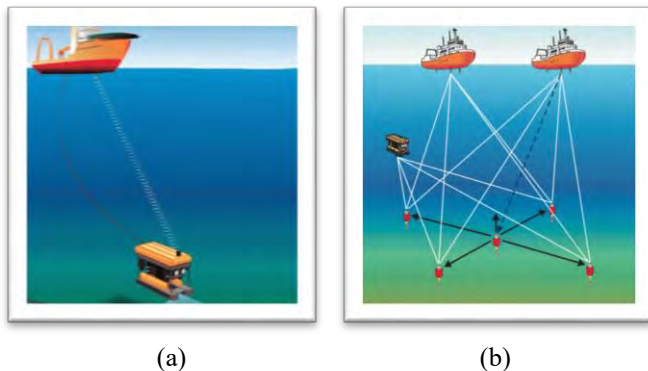
2.4. Acoustic Underwater Positioning

Peredaman sinyal radio di dalam air sangat tinggi, sehingga GPS tidak bisa digunakan untuk menentukan posisi objek di dalam air. Salah satu alternatif yang umum digunakan adalah *underwater acoustic positioning systems*. *Hydro-acoustic positioning* didasarkan pada prinsip dasar triangulasi, dan bekerja dengan mengukur waktu yang dibutuhkan suara yang dipancarkan pada suatu objek hingga kembali lagi. Dengan waktu yang terukur ini maka posisi dapat ditentukan. Terdapat dua metode untuk menentukan posisi dari objek yang menyelam, yaitu *Ultra Short Base Line (USBL)* dan *Long Base Line (LBL) systems*.

Sistem USBL menggunakan dua alat atau perlengkapan untuk mengukur posisi ROV, yaitu *interrogator* dan *transponder*. *Interrogator* merupakan kombinasi antara pemancar dan penerima yang mengirimkan sinyal akustik pada satu frekuensi dan menerima respon pada frekuensi lain. Alat ini dipasang pada kapal, sedangkan *transponder* dipasang pada *underwater robot*. *Transponder* mengirim sinyal respon ketika menerima sinyal dari *interrogator*. Jarak antara robot dan *interrogator* dihitung dengan mengukur waktu

tempuh sinyal melalui air. Sudut kemiringan dari sinyal respon dihitung dengan menggunakan *phase shift* antara beberapa *hydrophones* pada element *interrogator*. Kombinasi jarak dan sudut tersebut memungkinkan untuk mengetahui posisi dari ROV. Sistem USBL ditunjukkan oleh gambar 2.4(a).

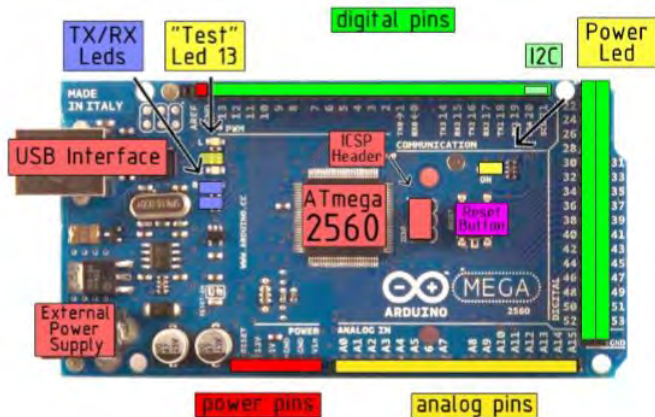
Sistem LBL berisi satu atau beberapa *interrogator*, dan satu set *transponder* lain yang disebut *pingers*. *Pingers* adalah alat yang secara kontinyu mengirim pulsa akustik dengan frekuensi spesifik. Pada sistem ini, *interrogator* terpasang pada ROV, dan *interrogator* memanfaatkan respon dari akustik penanda (*beacon*) lain untuk menghitung jarak ke masing-masing alat. Alat tersebut dipasang pada dasar laut dengan diketahui posisinya. Robot dengan *interrogator* menggunakan segitiga (*trilateration*) untuk menghitung posisinya sendiri relatif terhadap penanda. Sistem LBL ditunjukkan oleh gambar 2.4(b) [10].



Gambar 2.4 (a) Skema Sistem USBL; (b) Skema Sistem LBL [10].

2.5. Arduino Mega 2560

Arduino Mega 2560 adalah *board* mikrokontroler yang berbasis Atmega2560. Terdapat 54 pin digital *input/output* yang mana 14 pin diantaranya dapat digunakan sebagai pin *output* PWM, 16 pin *input* analog, sebuah *jack* daya, header ICSP, dan sebuah tombol reset. *Board* Arduino Mega 2560 ditunjukkan oleh gambar 2.5.



Gambar 2.5 Board Arduino Mega 2560 [11].

Untuk menggunakannya sangat mudah, cukup dengan menghubungkan dengan komputer melalui kabel USB atau menyalakan dengan adapter AC-to-DC atau bisa juga dengan baterai secara langsung. Spesifikasi teknik dari Arduino Mega 2560 ditunjukkan pada tabel 2.1 berikut.

Tabel 2.1. Spesifikasi Teknik Arduino Mega 2560 [11].

No	Parameter	Spesifikasi
1	Mikrokontroler	Atmega2560
2	Tegangan Operasi	5 V
3	Tegangan Input (Rekomendasi)	7-12 V
4	Tegangan Input (Batas)	6-20 V
5	Pin I/O Digital	54 (14 Pin PWM)
6	Pin Input Analog	16
7	Arus DC per pin I/O	40 mA
8	Arus DC untuk pin 3.3 V	50 mA
9	Flash Memory	256 KB (8 KB digunakan bootloader)
10	SRAM	8 KB
11	EEPROM	4 KB
12	Clock Speed	16 MHz

2.5.1. Sumber Daya/Power

Arduino Mega 2560 dapat diaktifkan melalui kabel USB atau dengan suplai daya eksternal. *Board* dapat beroperasi pada suplai eksternal dari 6 hingga 20 volt. Rentang tegangan operasi yang disarankan adalah 7 hingga 12 volt karena jika tegangan kurang dari 6 volt maka tidak akan stabil, sedangkan jika lebih dari 12 volt, maka dapat menyebabkan regulator tegangan menjadi panas dan kemungkinan paling buruk adalah dapat merusak *board* Arduino Mega.

2.5.2. Memori

Chip Atmega2560 yang terpasang pada Arduino Mega 2560 Revisi 3 memiliki memori 256 KB, dengan 8 KB dari memori tersebut telah digunakan untuk *bootloader*. Jumlah SRAM 8 KB dan EEPROM 4 KB, dimana pada memori ini dapat dilakukan baca-tulis dengan menggunakan EEPROM *library* saat melakukan pemrograman.

2.5.3. Input/Output

Pin *input/output* Arduino Mega 2560 jumlahnya yang terbanyak dari semua jenis *board* yang dikembangkan Arduino. Arduino Mega 2560 memiliki 54 pin digital yang dapat digunakan sebagai *input/output*. Pin-pin tersebut bekerja pada tegangan 5 volt. Pada Arduino Mega terdapat beberapa pin yang memiliki fungsi khusus, yaitu:

- Serial: Disini terdapat 4 serial yang masing-masing terdiri dari 2 pin, yakni RX dan TX. (a) Serial 0: pin 0 (RX) dan pin 1 (TX), (b) Serial 1: pin 19 (RX) dan pin 18 (TX), (c) Serial 2: pin 17 (RX) dan pin 16 (TX), (d) Serial 3: pin 15 (RX) dan pin 14 (TX). Pin RX adalah pin untuk *receive* atau menerima, sedangkan TX adalah *transmit* atau untuk mengirim data serial TTL. Pin 0 dan pin 1 adalah pin yang digunakan oleh chip USB-to-TTL Atmega16U2.
- Eksternal *Interrupt*: Pin *interrupt* yaitu pin 2 (*interrupt* 0), pin 3 (*interrupt* 1), pin 18 (*interrupt* 5), pin 19 (*interrupt* 4), pin 20 (*interrupt* 3), dan pin 21 (*interrupt* 2). Pin-pin tersebut dapat dikonfigurasi untuk men-*trigger* sebuah *interrupt* pada nilai “LOW”, tepi naik atau tepi turun, atau merubah nilai. Untuk mengatur *interrupt* tersebut, maka

- perlu menggunakan fungsi *attachInterrupt()*.
- PWM: Pin 2 hingga 13 dan 44 hingga 46 adalah pin-pin yang menyediakan *output* PWM 8-bit dengan menggunakan fungsi *analogWrite()*.
- SPI: Pin 50 (MISO), 51 (MOSI), 52 (SCK), dan 53 (SS) mendukung komunikasi SPI dengan menggunakan *SPI library*.
- LED: Pin 13, pin ini terhubung dengan *built-in* LED yang dikendalikan oleh pin digital nomor 13. Set HIGH untuk menyalakan LED, LOW untuk memadamkannya.
- I2C: Pin 20 (SDA) dan Pin 21 (SCL). Pin-pin ini mendukung komunikasi I2C (TWI) menggunakan *wire library*.

Selain pin-pin *digital*, juga terdapat pin-pin *analog* yang berjumlah 16 buah. Masing-masing pin *analog* tersebut memiliki resolusi 10 bits (rentang nilai dari 0 sampai 1024). Secara *default*, pin-pin ini diukur dari *ground* ke 5 volt, namun bisa juga dengan menggunakan fungsi *analogReference()*. Beberapa pin lainnya pada *board* ini yaitu:

- AREF: sebagai referensi tegangan *input analog*.
- Reset: Dengan menghubungkan ke “LOW” untuk melakukan reset terhadap mikrokontroler atau bisa juga dengan menekan tombol reset yang tersedia.

2.5.4. Komunikasi

Arduino Mega R3 dilengkapi fasilitas untuk berkomunikasi dengan komputer, arduino lainnya, ataupun dengan mikrokontroler lainnya. Chip Atmega 2560 menyediakan komunikasi serial UART TTL (5V) yang tersedia pada pin 0 (RX) dan pin 1 (TX). Chip Atmega16U2 pada *board* berfungsi untuk menerjemahkan bentuk komunikasi melalui USB dan akan tampil sebagai *virtual port* di komputer. *Firmware* 16U2 menggunakan driver USB standar sehingga tidak memerlukan driver tambahan.

Pada Arduino *Software* (IDE) terdapat serial monitor yang memudahkan data tekstual untuk dikirim menuju Arduino atau keluar dari Arduino. Ketika ada data yang ditransmisikan melalui Chip USB to serial via kabel USB ke komputer maka LED TX dan RX akan

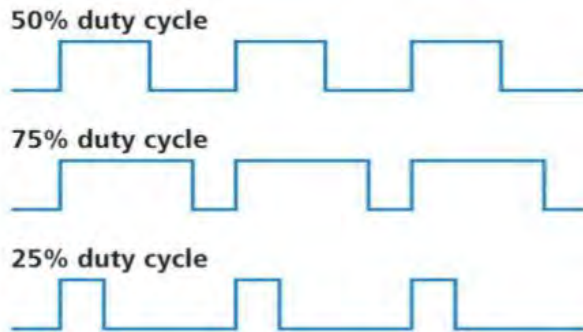
menyala berkedip-kedip. Pin digital juga dapat digunakan untuk komunikasi serial, yaitu dengan menggunakan *library SoftwareSerial*. Selain itu, Atmega2560 juga mendukung komunikasi I2C (TWI) dan SPI. Pada Arduino *Software* (IDE) telah terdapat *wire library* yang memudahkan pengguna menggunakan bus I2C, serta *SPI library* yang dipakai untuk menggunakan komunikasi SPI.

2.5.5. Perlindungan Arus USB

Arduino memiliki *polyfuse reset* yang melindungi *port* USB komputer pengguna dari arus pendek atau berlebih. Jika arus lebih dari 500 mA, maka *fuse* otomatis bekerja.

2.6. PWM (*Pulse Wide Modulation*)

Dalam aplikasi elektronika daya, dasar-dasar PWM banyak digunakan, yaitu untuk pengaturan konversi daya DC/DC, DC/AC, dan lain-lain. Selain aplikasi pada elektronika daya, yang umum juga digunakan pada pengaplikasian PWM adalah untuk pengendalian kecepatan motor DC, pengendalian motor servo, pengaturan nyala terang LED dan lain sebagainya. Secara sederhana, PWM merupakan sinyal dengan lebar pulsa bernilai “HIGH” dalam satu periode yang mewakili tegangan DC. Pulsa bernilai “HIGH” inilah yang menentukan nilai *duty cycle*. *Duty cycle* merupakan perbandingan lama waktu sinyal bernilai “HIGH” dalam satu periode (jumlah waktu “HIGH” dan “LOW”). Pulsa PWM dengan *duty cycle* yang berbeda-beda ditunjukkan oleh gambar 2.6.

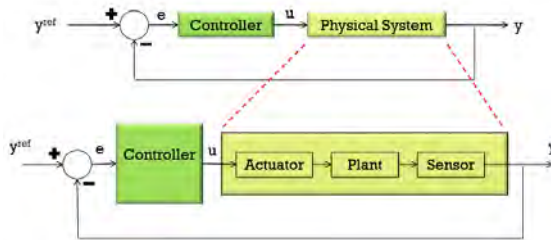


Gambar 2.6 *Duty Cycle* pada PWM [12].

2.7. PID (*Proportional – Integral – Derivative*)

Kontrol PID adalah suatu kontroler yang memiliki fungsi untuk menentukan kestabilan sistem instrumentasi dengan memberikan umpan balik pada sistem tersebut. Kontroler PID disusun dari tiga parameter, yakni *proportional* (P), *integral* (I), dan *differential* (D). Ketika diimplementasikan dalam sistem, parameter P, I, dan D dapat bekerja secara terpisah ataupun bekerja sebagai gabungannya, misal P dan I, P dan D ataupun ketiganya sekaligus sesuai dengan keinginan dan disesuaikan dengan *input* pada sistem [13].

Di setiap industri otomasi dan proses, kontrol PID selalu digunakan. Mulai dari industri kecil hingga industri dengan teknologi tingkat tinggi. Pada industri kilang minyak dan rancang bangun kapal, bekerja dengan PID adalah suatu rutinitas [14]. Algoritma kontrol PID umumnya menggunakan *feedback*. Skematik kontrol PID ditunjukkan oleh gambar 2.7.



Gambar 2.7 Skema PID Kontrol [15].

PID adalah kombinasi dari tiga blok pembentuk dasar, yaitu *proportional* (P), *integral* (I), dan *differential* (D). Pada kontrol *proportional*, rumusan matematika yang digunakan yakni:

$$P = K_p \times \text{Error} \quad (1)$$

Disini menggunakan proporsi *error* sistem untuk mengontrol sistem. Pada aksi ini, *offset* terdapat pada sistem.

Pada kontrol integral,

$$I = K_i \times \int \text{Error} \, dt \quad (2)$$

Disini merupakan jumlah proporsi pada *error* sistem. Aksi I akan membuat sistem sedikit tertinggal, karena akan mengimbangi respon yang diberikan oleh aksi P. Pada kontrol *differential* berlaku,

$$D = K_d \times (d(\text{Error})/dt) \quad (3)$$

Nilai D akan menyebabkan sistem memipin. Aksi D ini akan menyeimbangi *lag* sistem yang disebabkan oleh aksi I.

2.8. Driver Motor

Pada tugas akhir ini, *driver motor* yang digunakan adalah *driver motor* VN2SP30 yang ditunjukkan oleh gambar 2.8, sedangkan gambar 2.9 menunjukkan skematik rangkaian VN2SP30. VN2SP30 ini merupakan *driver motor* jembatan penuh yang ditujukan untuk berbagai macam aplikasi otomotif. Perangkat ini menggabungkan *driver* sisi tinggi monolitik ganda dan dua switch sisi rendah. *Input* sinyal INA dan INB dapat dihubungkan secara langsung ke mikrokontroler untuk memilih arah putaran motor dan menghentikan motor. Satu buah *driver* dapat mengontrol 1 buah PWM, dimana masing-masing PWM dignakan untuk mengontrol satu buah motor. Ringkasan dari spesifikasi *driver motor* ini dapat dilihat pada tabel 2.2 berikut [16].

Tabel 2.2. Spesifikasi *Driver Motor* VN2SP30 [16].

Simbol	Parameter	Nilai
Vcc	Tegangan Suplai	+41 V
Vcc	Tegangan Operasi	5.5-16 V
I _{max}	Arus keluaran maksimum (cont.)	30 A
I _R	Arus keluaran balik (cont.)	-30 A
I	<i>Practical Continuous Current</i>	14 A
f	<i>PWM frequency</i>	20 KHz
T _{TSD}	<i>Thermal shutdown temperature</i>	150-200 °C
T _{TR}	<i>Thermal reset temperature</i>	135 °C
R _{DS(on)}	<i>MOSFET on-resistance</i>	19 m (per leg)



Gambar 2.8 *Driver Motor* VN2SP30 [17].

Beberapa fitur pada *bilge pump motor* ini yaitu seperti dibawah ini:

- a. *Fully submersible*.
- b. Mudah dibersihkan.
- c. Diam dan tidak bergetar saat beroperasi.
- d. *Compact, efficient long life motor*.
- e. Perlindungan pengapian.
- f. Tidak terbakar saat dioperasikan pada kondisi kering.
- g. Perlindungan karat dan korosi.

Bilge pump motor yang digunakan beroperasi pada tegangan 12 volt dengan arus maksimal 3.3 A, serta dapat memompa hingga 1100 gph (gallons per hour) atau setara dengan 4000 liter per jam. Spesifikasi *bilge pump motor* ini disajikan dalam tabel 2.3 [19].

Tabel 2.3. Spesifikasi *Bilge Pump Motor* [19].

<i>GPH @ Open Flow</i>	1100 GPH
<i>Amp Draw @ Open Flow</i>	3.3 A
<i>Fuse Size</i>	6 amps
<i>Shaft Material</i>	<i>Stainless</i>
<i>Temperature Limit</i>	200 F
<i>Typical Life</i>	1500 Hours
<i>Wires</i>	2
<i>Wire gauge</i>	16 gauge
<i>Weight</i>	10 oz
<i>Height</i>	4-1/4"
<i>Width</i>	2-3/8"
<i>Depth</i>	2-3/8"

2.10. RC (Radio Control)

Radio control (RC) merupakan alat pengendali yang memanfaatkan gelombang radio. Penggunaannya untuk mengendalikan suatu kendaraan atau wahana lain dengan tangan secara *wireless*. Pemanfaatannya pun meliputi berbagai bidang, seperti industri, militer, keperluan penelitian, dan lain-lain. Sistem pada RC kebanyakan adalah sistem *open loop*, yang mana perintah dikirimkan melalui pemancar dan diterima oleh penerima. Pemancar mengirim data berdasarkan tombol yang siap memberikan sebuah masukan. Data dikirim melalui frekuensi gelombang radio sehingga penerima mengubahnya menjadi sebuah sinyal diskrit. Gambar 2.11

berikut merupakan perlengkapan RC [21].



(a)

(b)

Gambar 2.11 Perlengkapan RC (a) *DX7s RC Transmitter*; (b) *Orange RX Receiver* [21].

Model dari RC yang pertama adalah *single-channel self-built equipment*. Mengurangi penggunaan baterai dengan adanya transistor dan membutuhkan arus pada level tegangan yang rendah. Kontrolnya pun masih sederhana, hanya ada kanan, kiri, dan netral. Kedua yaitu *multi-channel*. Secara khusus untuk pesawat terbang dan minimum ada tiga dimensi yang dibutuhkan (*yaw*, menurun, dan kecepatan motor). Ada *Pulse-Position Modulation* (PPM) yang merupakan pengembangan dari *single-channel*. Terakhir, ada *Pulse-Code Modulation* (PCM) yang memberikan sinyal digital *bit-stream* terkomputisasi. Frekuensi yang banyak digunakan sebesar 2.4 GHz, namun ada juga RC yang menggunakan frekuensi 72 MHz [22].

2.11. Telemetri

Telemetri 433 MHz yang digunakan merupakan sistem radio telemetri yang berbasis sistem 3DR Radio dan kompatibel 100% jika dijalankan pada *board* dengan *firmware* yang sama. Selain 433 MHz, juga ada yang menggunakan kanal frekuensi di 915 MHz. Telemetri ini juga cocok dengan *ArduPilot* atau *Pixhawk-based systems* dan memungkinkan pengguna untuk menggunakan 2 telemetri sebagai pemancar dan penerima yang terintegrasi dengan mikrokontroler dan bisa dipasang pada *drone* dan *ground station*.



Gambar 2.12 *Telemetry Kit 433 MHz [23].*

Fitur-fitur dan spesifikasi telemetry yang ditunjukkan oleh gambar 2.12 diatas disajikan pada tabel 2.4 berikut.

Tabel 2.4. Spesifikasi Telemetry [24].

<i>Processing</i>	Daya <i>output</i> maksimum 100 mW
	Sensitivitas penerimaan -117 dBm
	Berbasis pada modul HopeRF's HM-TRP
	RP-SMA <i>Connector</i>
	Komunikasi 2 arah <i>full-duplex</i> dengan <i>adaptive TDM</i>
	Antarmuka UART
	<i>Transparent serial link</i>
	MAVLink <i>protocol framing</i>
	<i>Frequency Hopping Spread Spectrum</i> (FHSS)
	<i>Configurable duty cycle</i>
	Koreksi <i>error</i> hingga 25% dari bit <i>error</i>
	<i>Open-source SIK Firmware</i>
	Dapat dikonfigurasi dengan Mission Planner dan APM Planner
<i>Features</i>	Dapat ditukar antara modul udara dan <i>ground</i>
	915 atau 433 MHz
	Micro-USB <i>Port</i>
	6-Position DF13 <i>connector</i>
<i>Dimensions</i>	26.7 cm x 55.5 cm x 13.3 cm (tanpa antena)
<i>Power</i>	Tegangan suplai 3.7 – 6 VDC (dari USB atau DF13)
	Arus <i>transmit</i> : 100 mA pada 30 dBm
	Arus <i>receive</i> : 25 mA
	Antarmuka serial: 3.3 V UART

2.12. Global Positioning System (GPS)

GPS sebagai satu-satunya satelit navigasi global yang dapat menentukan lokasi, kecepatan, arah, dan waktu adalah alat yang banyak digunakan di bumi ini [25]. Salah satu model GPS yang ada yakni GPS U-Blox Neo M8N ditunjukkan oleh gambar 2.13.



Gambar 2.13 *U-Blox Neo M8N with Compass [26].*

Device diatas berisi GPS dan kompas digital HMC5883L. Oleh karenanya, pin pada *device* Neo M8N terdapat 6 pin, yakni pin vcc, *ground*, Rx, Tx, SDA, dan SCL. GPS diakses dengan menghubungkan pin Rx, Tx, sedangkan kompas dihubungkan dengan pin SDA dan SCL. Modul GPS NEO-M8 didesain untuk menerima dan melacak sinyal L1C/A pada frekuensi 1575.42 MHz dari GPS. Seri NEO-M8N menghasilkan sinyal dengan sensitivitas tinggi dan waktu minimum akuisisi meskipun pada daya rendah sistem. GPS ini dapat disuplai dengan tegangan 3.3 hingga 5 volt. Spesifikasi lebih lengkap GPS NEO-M8N disajikan dalam tabel 2.5 [27].

Tabel 2.5. Spesifikasi U-Blox Neo-M8N GPS Compass [27].

Suplai	2.7 hingga 3.6 Volt (disarankan)
Antarmuka	UART, USB, SPI, dan DDC (I2C compliant)
Fitur	Programmable (Flash)
	Data Logging
	Additional SAW
	Additional UNA
	RTC Crystal

2.13. Kompas

Kompas adalah alat yang digunakan untuk menentukan arah, mana selatan, mana utara, mana timur, dan mana barat. Kompas yang umum digunakan yaitu kompas analog [28]. Akan tetapi, selain kompas analog, juga ada kompas digital yang umum digunakan pada alat-alat elektronik. Salah satunya adalah kompas digital HMC5883L yang ditunjukkan oleh gambar 2.14.



Gambar 2.14 Kompas Digital HMC5883L [29].

Kompas digital HMC5883L ini merupakan *magnetometer* yang digunakan untuk menghitung kuat medan magnet dan menghitung arahnya. HMC5883L, didalamnya termasuk *high-resolution HMC118X series magneto-resistive sensors* yang dilengkapi penguat, pembatalan *offset*, dan 12-bit ADC dengan akurasi sudut 1° hingga 2° . Antarmuka kompas ini menggunakan I2C. Tegangan suplai berada pada rentang 3 hingga 5 volt (disarankan 3.3 volt) dan rentang pengukuran berada di antara 1.3 hingga 8 Gauss [30].

2.14. Photovoltaic (PV)

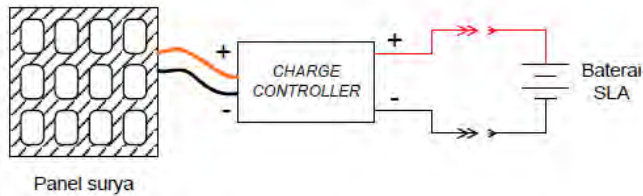
Photovoltaic (PV) atau biasa dikenal sebagai solar panel ataupun panel surya adalah bahan semikonduktor yang berfungsi untuk konversi cahaya menjadi listrik secara langsung. Beberapa bahan menunjukkan sebuah properti yang biasa dikenal sebagai efek fotolistrik yang menyebabkan beberapa bahan tersebut dapat menyerap foton cahaya dan melepaskan elektron. Ketika elektron bebas ini ditangkap, maka akan dihasilkan listrik yang dapat digunakan sebagai sumber listrik. Total daya keluaran PV sebanding dengan tegangan operasi dikalikan dengan arus operasi PV. Setiap PV memiliki *Maximum Power Point* atau biasa dikenal sebagai titik operasi. PV yang digunakan ditunjukkan pada gambar 2.15.



Gambar 2.15 *Photovoltaic (Solar Panel) 100Wp.*

2.15. MPPT Solar Charge Controller

Pemanfaatan PV dengan dipasangkan baterai selalu menggunakan sistem *charge controller*. Tujuannya agar penggunaan baterai dan energi listrik dapat optimal dan aman. *Charge controller* dapat mengatur tegangan minimal dan maksimal dari baterai sehingga dapat menjaga keamanan sistem. Yakni, pada pengisian berlebih dari PV, pemakaian berlebih oleh beban, mencegah arus balik ke PV, dan mencegah hubung singkat pada beban. Gambar 2.16 menunjukkan konsep dasar *charge controller* [31].



Gambar 2.16 Konsep *Charge Controller* [31].

Tipe *Charge Controller* yang digunakan pada tugas akhir ini adalah MPPT T40 yang ditunjukkan pada gambar 2.17. Tipe ini merupakan penyempurnaan dari tipe sebelumnya, yaitu M40. MPPT memiliki kepanjangan *Maximum Power Point Tracking*. Dengan MPPT T40 ini panel surya dapat beroperasi pada tegangan ideal panel surya itu sendiri dengan menghasilkan daya maksimum yang tersedia.

Kontroler ini memiliki tiga waktu sebagai pembagian fungsi, yaitu malam sebagai waktu kerja, jeda waktu, dan waktu kerja *dawn* (pagi). Selain itu, kontroler ini hemat daya saat *stand by*, yakni konsumsi daya hanya 10 mA – 15 mA. Fungsi perlindungan yang komprehensif juga tertanam pada kontroler MPPT T40, sehingga

seluruh sistem dapat beroperasi secara efisien dan aman. Fitur lain pada MPPT ini adalah sensor suhu pada kontroler ini.

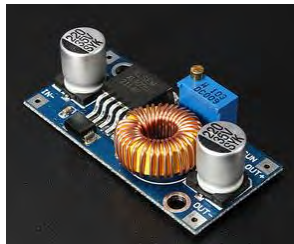
Konektor yang berada pada kontroler ada tiga pasang, pertama untuk masukan daya dari solar panel, kedua keluaran untuk mengisi daya baterai, dan ketiga keluaran menuju beban yang akan disuplai.



Gambar 2.17 MPPT Solar Charge Controller T40.

2.16. Adjustable Buck Converter

Tegangan suplai arduino berada pada 7 – 12 volt, sehingga diperlukan sebuah penurun tegangan agar tegangan yang diberikan pada arduino tidak melebihi ataupun kurang dari yang dibutuhkan. Dalam hal ini digunakanlah sebuah *Buck Converter* yang menggunakan IC XL4005. Tegangan masukan adalah mulai dari 5 hingga 32 volt, dengan tegangan masukan ini, tegangan keluaran yang dihasilkan berada pada rentang 0.8 hingga 30 volt. Dimana, tegangan keluaran dapat diatur oleh pengguna dengan memutar *multiturn* yang ada pada *board buck converter*. Arus keluaran maksimum sebesar 5 A. Kecepatan *switching* frekuensinya yaitu pada 300 KHz [32]. Gambar 2.18 berikut menunjukkan gambar 5A DC-DC *Adjustable Buck Converter*.



Gambar 2.18 Adjustable Buck Converter [33].

2.17. Baterai Lipo (*Litium Polimer*)

Baterai Lipo sudah umum digunakan sebagai sumber pada sebuah rangkaian elektronika. Didalamnya terdapat elektrolit polimer kering yang berbentuk seperti lapisan plastik film yang tipis. Lapisan film ini disusun berlapis-lapis diantara anoda dan katoda sehingga dapat mengakibatkan terjadinya pertukaran ion. Metode ini memungkinkan Lipo untuk dibuat dalam berbagai bentuk dan ukuran. Namun, karena terdapat sekat yang berupa elektrolit polimer kering yang membatasi antara anoda dan katoda membuat aliran pertukaran ion menjadi lemah. Hal ini berakibat pada penurunan *charging* dan *discharging rate*. Solusi untuk masalah ini yaitu dengan memanaskan baterai, sehingga pertukaran ion akan menjadi lebih cepat. Akan tetapi, masalah keamananlah yang berbicara, dimana memanaskan baterai dapat membuat baterai terbakar atau bahkan meledak. Baterai Lipo memiliki kelebihan sekaligus kekurangan jika dibandingkan baterai jenis lain seperti NiCd atau NiMH seperti yang disebutkan dalam tabel 2.6 berikut. Pada tugas akhir ini, menggunakan baterai lipo 3 sel yang ditunjukkan pada gambar 2.19 [34].

Tabel 2.6. Kelebihan dan Kekurangan Baterai Lipo [34].

Kelebihan	Kekurangan
Bobot ringan dan tersedia dalam berbagai bentuk dan ukuran.	Harga relatif mahal.
Kapasitas penyimpanan energi listik besar.	<i>Lifecycle</i> baterai Lipo pendek, sekitar 300-400 kali siklus pengisian ulang, namun juga bergantung pada perlakuan yang diberikan pada baterai.
Tingkat <i>discharge rate</i> energi tinggi.	Alasan keamanan, Lipo memakai bahan elektrolit yang mudah terbakar.



Gambar 2.19 Baterai Lipo 3 Sel.

2.18. AutoCAD 2015 (Student Version)

AutoCAD merupakan *software* CAD yang memiliki fungsi untuk menggambar atau mendesain sebuah objek 2 dimensi maupun 3 dimensi dengan sistem CAD. *Software* ini diproduksi oleh Autodesk yang dimulai dari versi pertama 1.0 yang diluncurkan pada tahun 1982. Seiring perkembangannya banyak penambahan fitur yang terjadi untuk melengkapi kebutuhan desain bagi para *designer*. Logo AutoCAD ditunjukkan oleh gambar 2.20.



Gambar 2.20 Logo AutoCAD [35].

AutoCAD banyak digunakan oleh insinyur sipil, arsitek, listrik, mesin, dan beberapa pekerjaan jenis lainnya. AutoCAD memiliki keunggulan yaitu berupa pengerjaan yang relatif singkat dengan kualitas hasil gambar yang baik. Oleh karena itu, aplikasi AutoCAD banyak ditemui di dunia industri [36].

2.19. Tinjauan Pustaka

Tinjauan pustaka bertujuan untuk membandingkan perangkat yang telah ada dan dikembangkan sebelumnya dengan perangkat yang dirancang pada tugas akhir ini. Berikut merupakan judul paper atau proyek yang dibandingkan dengan proyek pada tugas akhir ini.

2.19.1. *Underwater Communication Systems: A Review* [9]

Dalam *review Underwater Communication System* disebutkan bahwa komunikasi bawah air terdapat dua metode, salah satunya yaitu dengan skema sistem komunikasi buoyant. Dimana metode inilah yang digunakan dalam tugas akhir ini. Selain mengadopsi metode komunikasi buoyant, juga dilakukan pengembangan pada robot yang dibuat. Yakni, robot dapat dikendalikan melalui *remote control*. Sistem kendalinya pun terdapat tiga mode, yaitu mode manual, mode *auto*, dan mode loiter. Untuk keperluan daya, daya disuplai oleh tiga modul panel surya berkapasitas 100 Wp.

2.19.2. *Solar Powered Ferry Boat for the Rural Area of Bangladesh* [37]

Pada paper ini, dikembangkan sebuah kapal ferry bertenaga surya. Kapal bertenaga surya ini dibuat dengan tujuan untuk mengurangi polusi jika menggunakan bahan bakar minyak. Sistem kendali pada kapal ini adalah secara manual, yaitu dikendalikan secara langsung oleh pengendara kapal dengan menggerakkan *rudder* kapal. Perbedaan dengan *Buoyant Boat* yang dikembangkan pada tugas akhir ini yaitu pada kendali dan jumlah motor yang digunakan. Dimana, kendali *Buoyant Boat* menggunakan *remote control* dan motor yang digunakan berjumlah 2 motor sebagai penggerak utamanya.

2.19.3. *Autonomous Surface Vehicle for Measuring Water Body Parameters* [38]

ASV atau *Autonomous Surface Vehicle* yang dikembangkan pada paper ini bertujuan untuk menjaga air dengan menghimpun data dari parameter-parameter air yang dibutuhkan, seperti pH dan sebagainya. ASV ini juga menggunakan GPS dan kompas sebagai navigasi. Terdapat mode manual dan mode autonomous yang akan bergerak mengikuti titik-titik lokasi yang ditentukan. Perbedaan dengan *Buoyant Boat* pada tugas akhir ini yaitu *Buoyant Boat* menggunakan panel surya untuk memenuhi kebutuhan daya dan pada sistem kendali terdapat mode loiter, yang mana pada paper ini tidak terdapat mode ini.

2.19.4. *Evolutionary Path Planning of an Autonomous Surface Vehicle for Water Quality Monitoring* [39]

Evolusi pada perencanaan rute yang dimaksud disini ialah penentuan rute perjalanan dengan menggunakan Genetic Algorithm. Jadi pergerakan kapal mengikuti logika yang diperoleh dari hasil Genetic Algorithm yang menghasilkan solusi optimal. Sesuai dengan judul paper, dimana tujuan utama kapal ini dibuat yaitu untuk monitoring kualitas air. Perbedaan dengan *Buoyant Boat* pada tugas akhir ini yaitu pada *Buoyant Boat* terdapat tiga sistem kendali, manual, auto, dan loiter. Selain itu, *Buoyant Boat* juga dilengkapi dengan panel surya.

2.19.5. *Design of Small Autonomous Boat for Course-keeping Manuvers* [40]

Kapal disini dibuat dengan memodifikasi kapal mainan yang ada dipasaran dan sistem didalamnya diganti dengan sistem tertanam, sehingga dapat dikontrol dengan sebuah *remote control* (RC). Kapal ini dilengkapi dengan sensor (kompas digital) untuk menghasilkan manuver yang simpel. Untuk menggerakkan kapal, digunakan sepasang motor dan *rudder*. Dibandingkan dengan *Buoyant Boat*, sistem kendali pada kapal hanya berupa kendali manual dan *autonomous*. Sedangkan, pada *Buoyant Boat*, terdapat tambahan kendali, yakni kendali loiter, dimana fungsi ini digunakan untuk menjaga posisi agar tetap pada lokasi yang telah dikunci (*lock location*).

2.19.6. *Development of Unmanned Surface Vehicle for Smart Water Quality Inspector* [41]

Kapal ini dibuat dengan tujuan untuk memeriksa kondisi air. Dalam pengambilan data kondisi air menggunakan *wireless sensor network* yang dapat mengakuisi data secara *real-time*. Untuk membuat rute pergerakan kapal, digunakanlah sebuah GPS. Dimana kapal dapat bergerak berdasarkan titik-titik yang telah di atur sebelumnya. Kendali pada kapal ini ada dua, yaitu secara manual dan *autoomous*. Dengan *Buoyant Boat*, kapal yang dikembangkan pada paper ini berbeda, yaitu pada kendali *Buoyant Boat* terdapat mode *loiter*, selain kendali manual dan *autonomous*. Sedangkan, motor yang digunakan sama, yaitu dua motor, satu motor kanan, dan satu motor kiri.

2.19.7. *Green Sailing of Solar PV Powered Country Boat using Buck-Boost Chopper* [42]

Kapal disini, dibuat untuk mengurangi polusi yang diakibatkan oleh mesin diesel pada sebuah kapal. Oleh karena itu, kapal yang dikembangkan pada paper ini menggunakan panel surya sebagai sumber daya. Fokus kapal pada paper ini yaitu pada performa PV dan peningkatan efisiensinya. Sedangkan, dibandingkan dengan *Buoyant Boat*, PV pada *Buoyant Boat* tidak ada metode untuk peningkatan efisiensi, tetapi PV hanya digunakan sebagai pemenuhan daya. Namun, jika pada kapal yang dipaper ini tidak menggunakan sistem kendali dengan *remote control*, pada *Buoyant Boat* memanfaatkan *remote control* dan memiliki tiga mode kendali.

2.19.8. *Research Solar Power Boat – Data Management and Online Visualization* [43]

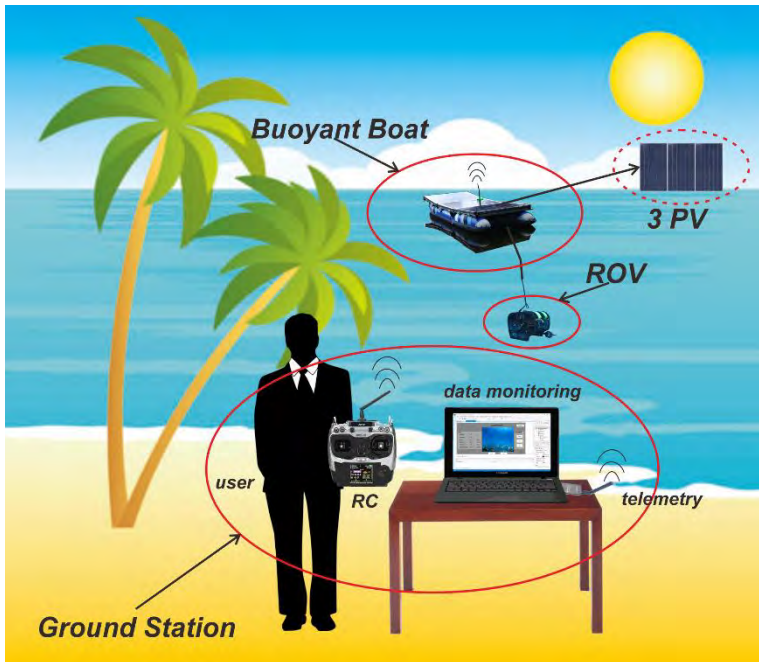
Kapal disini menggunakan PV sebagai sumber tenaga. Kapal yang digunakan adalah kapal yang dapat ditumpangi pengemudi dan dapat dikemudikan secara langsung. Fokus pada kapal ini yaitu untuk mengetahui informasi daya pada baterai yang bersumber dari PV menggunakan IMS (*Information-Management-System*). Perbedaan dengan *Buoyant Boat*, yakni *Buoyant Boat* dikendalikan menggunakan *remote control* dan memiliki tiga mode kendali. Penggunaan PV pada *Buoyant Boat* hanya sebatas pemanfaatan sebagai sumber tenaga pada *Buoyant Boat*.

2.19.9. *Soft-Switching PWM Forward Power Converter Auxiliary Active Clamped Capacitor for Solar Energy-Driven Boat* [44]

Pada paper ini, dikembangkan sebuah kapal yang memanfaatkan PV sebagai sumber daya utama. Fokus pada paper ini yakni pada pemanfaatan PV, tidak pada sistem kendali kapal. Sehingga, dibandingkan dengan *Buoyant Boat*, terdapat perbedaan pada fitur sistem kendali dan pemanfaatan PV. Dimana, sistem kendali pada *Buoyant Boat* ada tiga, sedangkan pada kapal di paper ini hanya kendali manual. Selain kendali, pemanfaatan PV pada *Buoyant Boat* hanya sebatas digunakan, tanpa ada metode khusus untuk meningkatkan efisiensi dari PV yang digunakan.

BAB III

PERANCANGAN SISTEM



Gambar 3.1. Skema Sistem Keseluruhan.

Rancang bangun *Buoyant Boat* ini tergabung dalam sistem besar, yang terdiri dari *Ground Station*, *Buoyant Boat*, dan *ROV* seperti yang ditunjukkan oleh gambar 3.1 diatas. *Buoyant Boat* dengan *Ground Station* terhubung secara *wireless*, sedangkan *Buoyant Boat* dengan *ROV* terhubung melalui sebuah kabel. Pengguna yang berada di *Ground Station* dapat mengoperasikan *Buoyant Boat* dan *ROV* dari jarak jauh dengan mengamati *display monitor*, dimana pada *display monitor* juga ditampilkan data yang dikirim oleh *ROV* maupun *Buoyant Boat* sendiri. Data yang dikirim berupa data dan video. *Buoyant Boat* untuk membantu komunikasi antara *ROV* dengan *Ground Station* dilengkapi dengan tiga mode kendali, manual, *auto*, dan *loiter*. Serta, karena tujuannya agar dapat dioperasikan dalam jangka waktu yang relatif lama, maka *Buoyant Boat*

dilengkapi dengan 3 PV untuk memenuhi kebutuhan daya. Robot bawah air, ROV, juga dapat dioperasikan dari jarak jauh dengan bantuan *Buoyant Boat*. Ketika ROV akan melakukan penyelaman, *Buoyant Boat* berada pada mode loiter sehingga *Buoyant Boat* akan terus berada pada area tersebut tanpa khawatir terbawa arus karena *Buoyant Boat* akan kembali ke titik lokasi tersebut. Terlepas dari sistem besar, pada tugas akhir ini hanya membahas *Buoyant Boat* dan *Ground Station*, tanpa ikut serta dalam perancangan ROV. Spesifikasi *Buoyant Boat* yang dirancang pada tugas akhir ini disajikan pada tabel 3.1 berikut:

Tabel 3.1. Spesifikasi *Buoyant Boat*.

Dimensi	255 x 110 x 30 cm
Berat Total	54 Kg
Jumlah Motor	2 Buah
Jumlah Panel Surya	3 Modul @100Wp
Fitur Mode Kendali	Manual, Auto, Loiter
Fitur Monitoring Data	Display data dan video

Pada bab perancangan sistem ini menjelaskan sistem secara keseluruhan mulai dari perangkat keras hingga perangkat lunak yang dibutuhkan untuk fungsi-fungsi yang akan dirancang pada *Buoyant Boat*. Pada *Buoyant Boat* dirancang fungsi kendali yang terdiri dari tiga mode kendali: manual, auto, dan loiter. Selain itu, juga dirancang penggunaan panel surya dan perancangan *graphical user interface (GUI)* untuk menampilkan data dari *Buoyant Boat* ke *Ground Station*. Perangkat keras yang digunakan antara lain, Arduino Mega 2560, *driver motor* VNH2SP30, motor DC, GPS, kompas, *buck-converter*, sepasang remot kontrol dengan *receiver* AT9 Radiolink, modul panel surya, MPPT *Solar charge controller*, baterai lipo 3 sel, serta perahu karet sebagai badan kapal. Sedangkan, perangkat lunak terdiri dari *software* Arduino (IDE) untuk pembuatan kode program dan AutoCAD 2015 *Student Version* untuk membuat desain-desain yang diperlukan *Buoyant Boat* dengan menggunakan 3D printer.

Perancangan perangkat keras meliputi perancangan mekanik dan perancangan komponen elektronik. Perancangan mekanik terdiri dari pembuatan dudukan panel surya pada badan kapal dan pembuatan dudukan komponen elektronik yang akan diletakkan pada kapal. Perancangan komponen elektronik antara lain meliputi kombinasi sensor, *driver motor*, dan remot kontrol dengan mikrokontroler.

Sedangkan, perancangan perangkat lunak meliputi proses akuisi data

remot kontrol, sensor GPS dan kompas, perancangan algoritma kendali (manual, *auto*, dan *loiter*), penggunaan telemetri, dan pembuatan desain-desain perangkat keras yang diperlukan dengan menggunakan AutoCAD, untuk kemudian di cetak dengan printer 3D.

3.1. Diagram Blok Sistem

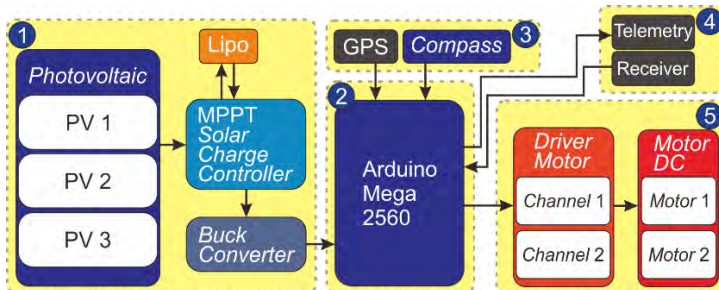
Seperti dijelaskan sebelumnya, bahwa secara umum sistem menggunakan Arduino Mega 2560 sebagai pengolah data. Suplai daya yang dibutuhkan sistem bersumber dari tiga modul panel surya yang mana daya disimpan pada baterai Lipo melalui sebuah MPPT *Solar Charge Controller*. MPPT *Solar Charge Controller* memiliki logika pengisian daya yang dapat diatur sendiri oleh pengguna. Pada MPPT yang digunakan terdapat tiga pengaturan, yaitu untuk *PV Off*, *Load Off*, dan *Load On*. *PV Off* digunakan untuk menghentikan proses pengisian daya dari PV ke baterai. Pada tugas akhir ini, *PV Off* di atur pada tegangan 12.6 volt, sehingga ketika tegangan baterai sudah mencapai 12.6 volt, maka secara otomatis MPPT akan menghentikan proses pengisian daya baterai dan tegangan yang dikeluarkan pada beban adalah tegangan yang langsung dari PV, bukan dari dari baterai lagi. *Load Off* diatur pada tegangan 10.5 volt. Hal ini bertujuan untuk menghentikan daya yang keluar ke beban dan kemudian hanya berfokus untuk melakukan pengisian daya baterai. *Load On* diatur pada tegangan 12.0 volt. *Load On* akan mengaktifkan kembali daya yang mengalir ke beban ketika tegangan baterai sudah mencapai 12.0 volt dari keadaan sebelumnya pada posisi *Load Off*. Dengan regulasi diatas, proses pengisian daya pada Lipo akan menjadi aman.

Sistem kendali pada *Buoyant Boat* terdapat tiga mode, pertama mode kendali manual, kedua mode *auto*, dan ketiga mode *loiter*. *Switch* mode kendali dilakukan dengan remot kontrol. Kendali manual dikendalikan penuh melalui remot kontrol dengan mudah yang dilakukan oleh pengguna. *Buoyant Boat* dapat bergerak ke segala arah, maju, mundur, kiri, kanan, serong maju, dan serong mundur. Kendali mode *auto* akan membuat *Buoyant Boat* bergerak sesuai rute yang telah ditetapkan sebelumnya pada program. Pembuatan rute diatur dengan menetapkan titik-titik koordinat yang akan dilalui pada program. Jumlah titik-titik koordinat juga dapat diatur sesuai keinginan. Pada pengujian tugas akhir ini, digunakan tiga titik koordinat yang berada di danau delapan ITS. Terakhir

kendali mode loiter. Maksud kendali loiter disini yakni mempertahankan posisi *Buoyant Boat* pada lokasi yang telah di kunci dengan dikendalikan oleh remot kontrol. *Buoyant Boat* akan kembali ke titik yang telah dikunci jika *Buoyant Boat* berpindah sejauh 2 meter.

Kendali dari remot kontrol adalah secara *wireless*. Data diterima oleh *receiver* yang terhubung dengan Arduino Mega 2560. Kemudian data diolah oleh Arduino dan didefinisikan menjadi kontrol yang diinginkan oleh pengguna, berupa *switching* mode kendali ataupun kendali *Buoyant Boat* secara manual dan *lock location* untuk kendali loiter. Pada *Buoyant Boat* juga dipasang telemetri yang pasangannya terpasang pada laptop. Telemetri digunakan untuk mengirim data yang diambil pada *Buoyant Boat* ke monitor, sehingga pengguna dapat mengamati data-data yang diperlukan pada komputer pengguna.

Keseluruhan blok diagram ditunjukkan oleh gambar 3.2 dibawah ini.



Gambar 3.2. Diagram Blok Sistem.

Bagian-bagian pada blok diagram diatas terdiri dari:

- (1) Suplai Daya
 Daya yang dibutuhkan sistem disuplai dari tiga modul panel surya yang diregulasi dengan MPPT *solar charge controller* untuk kemudian disimpan dalam baterai dan dikeluarkan menuju beban.
- (2) Mikrokontroler
 Data dari sensor, telemetri, dan *receiver* diolah menggunakan mikrokontroler Arduino Mega 2560.
- (3) Sensor
 Keperluan navigasi *Buoyant Boat* menggunakan sensor kompas dan GPS.

(4) Komunikasi

Komunikasi kendali dari remot kontrol ke *receiver* yang terletak di *Buoyant Boat* dan komunikasi data dari *Buoyant Boat* ke *ground station* menggunakan sepasang telemetri.

(5) Aktuator

Komponen aktuator yang terdiri dari dua *driver motor* dan dua motor adalah sebagai penggerak bagi *Buoyant Boat*. Motor digerakkan sesuai data yang diperoleh dari sensor kompas dan GPS.

3.2. Perancangan Perangkat Keras

Perangkat keras pada tugas akhir ini antara lain: desain mekanik dan elektronik. Desain mekanik meliputi penempatan panel surya, motor beserta *propeller* yang digunakan, dan peletakan komponen elektronik. Sedangkan, desain elektronik meliputi perakitan semua komponen-komponen elektronik yang digunakan.

3.2.1. Perahu Karet Sebagai Kapal

Kapal yang digunakan adalah perahu karet yang umumnya digunakan untuk keperluan saat banjir ataupun untuk menikmati indahnnya pantai. Perahu karet yang digunakan memiliki dimensi 255 x 110 cm saat sudah diisi angin, sedangkan berukuran 264 x 123 saat belum diisi angin. Kapasitas berat maksimum perahu karet ini yakni 190 Kg atau setara dengan 2 orang dewasa. Biasanya, perahu karet ini ditambah sebuah motor khusus perahu karet agar bisa dinaiki dan dijalankan tanpa perlu mendayung. Perahu karet tersebut ditunjukkan oleh gambar 3.3 dibawah ini.

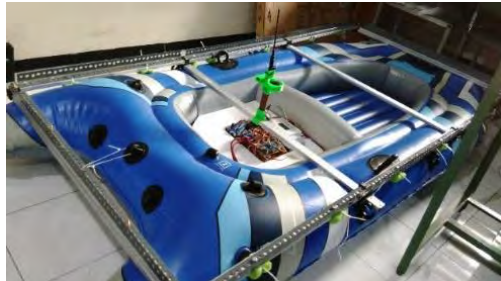


Gambar 3.3. Perahu Karet [45].

3.2.2. Tata Letak Panel Surya

Perancangan tempat panel surya pada kapal sebagian besar menggunakan aluminium dan besi siku seperti terlihat pada gambar 3.4. Terdapat tiga tempat yang disediakan untuk tiga modul panel surya. Pemotongan aluminium dan besi siku dilakukan di

laboratorium a206 menggunakan gergaji besi. Sedangkan, konektor yang menghubungkan masing-masing aluminium dan besi siku menggunakan hasil cetakan dari printer 3D dengan bahan ABS (*Acrylonitrile Butadiene Styrene*). Filamen ABS memiliki keunggulan tahan panas dan lebih kuat jika dibandingkan dengan filamen PLA (*Polylactic Acid*).



Gambar 3.4. Perancangan Tempat Panel Surya.

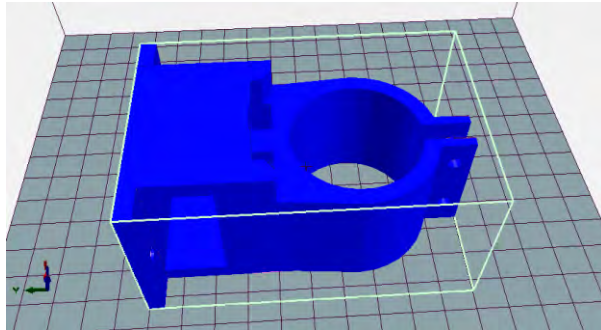
Aluminium dan besi siku yang digunakan direkatkan dengan kapak menggunakan kabel *ties*. Untuk menghindari kebocoran pada kapal akibat penggunaan aluminium dan besi siku maka pada bagian rawan bocor dilapisi dengan gabus silinder yang telah dipotong-potong. Setelah dudukan panel surya siap, maka panel surya siap dipasang pada kapal. Hasil pemasangan tiga panel surya pada kapal ditunjukkan oleh gambar 3.5.



Gambar 3.5. Pemasangan Tiga Modul Panel Surya Pada *Buoyant Boat*.

3.2.3. Posisi Motor

Buoyant Boat menggunakan dua buah motor DC 12 volt (*bilge pump motor*). Dua motor tersebut diletakkan dibagian bawah kapal dengan jarak antar motor sejauh 70 cm. Posisi motor diukur dari bagian depan kapal adalah berjarak 180 cm, dan dari belakang berjarak 75 cm. Penyangga motor menggunakan aluminium, sedangkan dudukan motor dicetak menggunakan printer 3D. Desain dudukan motor ditunjukkan oleh gambar 3.6, sedangkan peletakan motor pada kapal ditunjukkan oleh gambar 3.7.



Gambar 3.6. Desain Tempat (Dudukan) Motor.



Gambar 3.7. Posisi Motor pada Kapal.

3.2.4. Desain *Propeller*

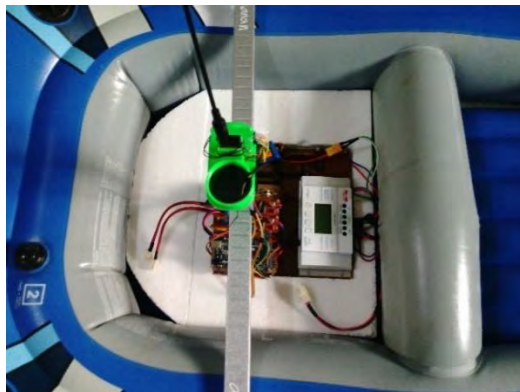
Propeller yang digunakan memiliki diameter 6 cm. Filamen ABS digunakan untuk mencetak *propeller* dengan 3D printer. *Propeller* yang telah dicetak juga dilapisi dengan resin dan dilumuri dengan lem G (sejenis lem alteco) agar lebih kuat dan lebih baik dari segi estetika. Gambar 3.8 berikut menunjukkan *propeller* yang digunakan pada tugas akhir ini.



Gambar 3.8. *Propeller*.

3.2.5. Tata Letak Komponen Elektronik Utama

Komponen elektronik utama pada *Buoyant Boat* terdiri dari Arduino Mega 2560, GPS, kompas, *driver motor*, *buck-converter*, MPPT *solar charge controller*, *fuse*, dan saklar. Semua komponen elektronik ini diletakkan dalam satu tempat yang sama menggunakan *acrylic* dan *spacer PCB*. Komponen elektronik utama ini diletakkan pada badan kapal seperti ditunjukkan pada gambar 3.9.



Gambar 3.9. Komponen Elektronik Utama.

Pada gambar 3.9 terlihat *styrofoam* gabus berwarna putih. *Styrofoam* gabus tersebut berguna untuk menjaga komponen elektronik utama dari guncangan. Sehingga, ketika guncangan terjadi pada kapal, posisi komponen elektronik tetap aman. Kelebihan lain dari *styrofoam* gabus tersebut yaitu dengan tingkat kelenturannya mampu membuat benturan-benturan kecil menjadi tidak berarti dan dapat diabaikan.

3.2.6. Tata Letak Komponen Elektronik Komunikasi

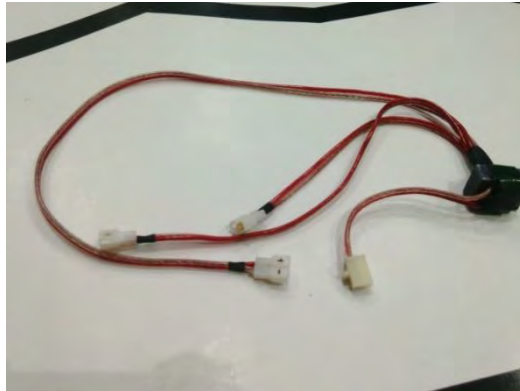
Buoyant Boat menggunakan tiga modul panel surya yang ketika dipasang pada kapal dapat menutupi seluruh bagian badan kapal. Oleh karena itu, diperlukan desain tempat GPS, *receiver* dan telemetri agar sinyal yang akan diterima maupun dipancarkan dari komponen ini tidak tertutup oleh panel surya. Desain tempat ketiga komponen tersebut ditunjukkan oleh gambar 3.10. Tempat tiga komponen komunikasi didesain menggunakan *software* AutoCAD dan dicetak dengan 3D *printer*. Filamen yang digunakan adalah filamen PLA karena desain ini cukup detail.



Gambar 3.10. Desain Peletakan Komponen Elektronik Komunikasi.

3.2.7. Panel Surya

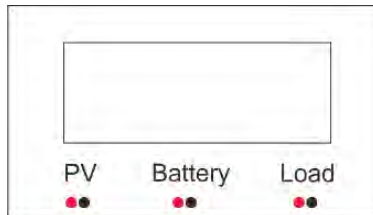
Terdapat tiga modul panel surya yang digunakan. Ketiga modul ini disusun secara paralel, sehingga total arus merupakan jumlah dari arus yang dihasilkan masing-masing panel surya. Sedangkan, tegangan tetap pada 13.8 volt sesuai spesifikasi dari panel surya 100 Wp yang digunakan. Penyambungan panel surya menggunakan terminal dengan 3 pasang pin yang ditunjukkan pada gambar 3.11.



Gambar 3.11. Terminal Penghubung.

3.2.8. MPPT Solar Charge Controller

Arsitektur MPPT yang digunakan ditunjukkan oleh gambar 3.12 berikut.

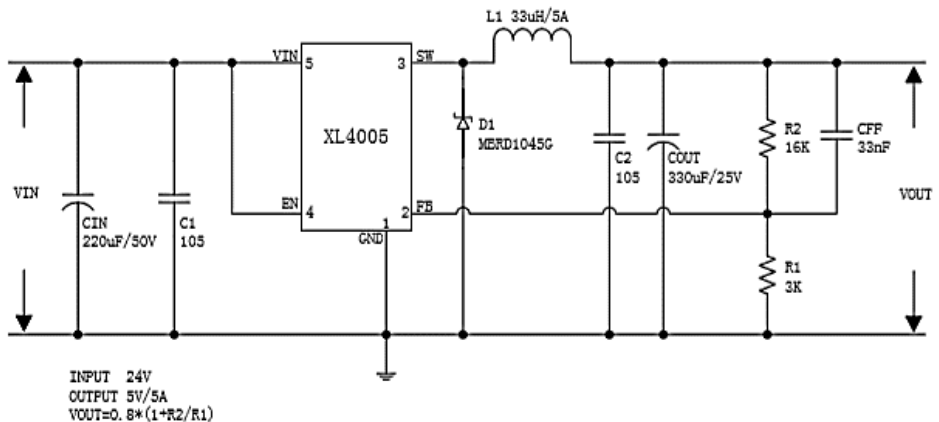


Gambar 3.12. Konfigurasi pada MPPT Solar Charge Controller.

Pada gambar diatas, terdapat tiga pasang terminal, masing-masing untuk dihubungkan dengan panel surya, baterai dan dihubungkan sebagai suplai daya ke beban. MPPT ini sudah memiliki regulasi pengisian daya sendiri. Regulasi tersebut berguna untuk menjaga keamanan dalam proses pengisian daya baterai.

3.2.9. Buck-Converter

Buck converter merupakan sebuah regulator tegangan yang dapat menurunkan tegangan 24 volt hingga mendekati 0 volt. Pada *Buoyant Boat*, tegangan yang diturunkan dari 12 volt ke 5 volt digunakan sebagai suplai daya untuk Arduino yang dihubungkan langsung ke pin 5 volt dan GND pada Arduino. Tegangan keluaran dari *Buck Converter* ini stabil, sehingga aman bagi mikrokontroler. Sumber 12 volt berasal dari keluaran MPPT *Solar Charge Controller* yang terhubung melalui pin *Load*. Rangkaian *Buck Converter* yang digunakan pada tugas akhir ini ditunjukkan oleh gambar 3.13.

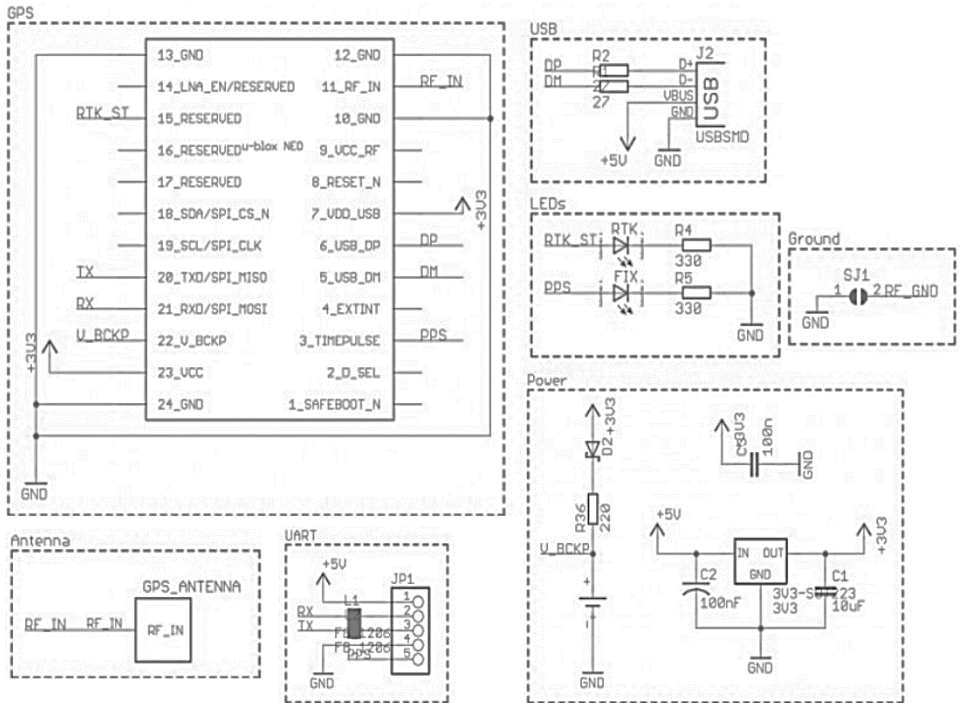


Gambar 3.13. XL4005 Typical Application Circuit (24V~5V/5A) [32].

3.2.10. Arduino Mega 2560

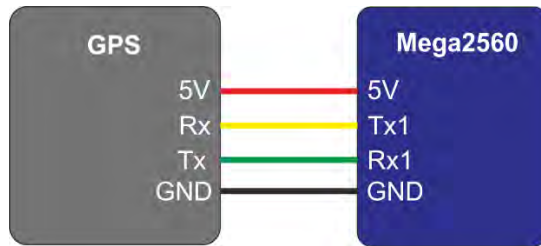
Pada tugas akhir ini arduino digunakan sebagai pengolah data sensor dan remot kontrol. Data-data yang diterima tersebut kemudian diubah menjadi nilai pergerakan *Buoyant Boat*. Dibutuhkan tegangan 5 volt untuk mengoperasikan mikrokontroler ini. Pada sistem ini digunakan pin digital sebanyak 6 pin (pin 4, pin 5, pin 6, pin 7, pin 9, dan pin 10), pin analog sebanyak 4 pin (pin A2, pin A3, pin A4, pin dan A5), pin *transmit* dan *receive* (TX0, RX0, dan TX1, RX1), pin I2C (SDA dan SCL). Selain itu, pin 5 volt dan pin GND juga digunakan, baik sebagai pin suplai daya untuk mikrokontroler maupun sebagai *output* ke elektronik lain. Skematik rangkaian Arduino Mega 2560 dapat dilihat pada gambar 3.14 yang terdapat pada halaman selanjutnya.

3.2.11. GPS



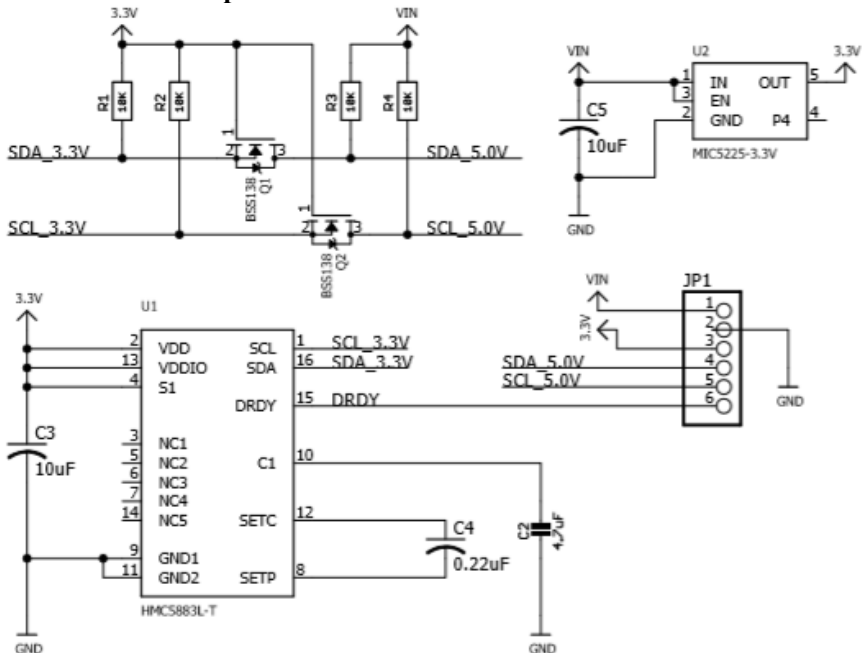
Gambar 3.15. Skematik GPS U-Blox M8N [47].

GPS merupakan navigator utama pada *Buoyant Boat*. Data-data dari satelit yang diterima GPS diteruskan menuju Arduino untuk diolah. Data GPS yang digunakan yakni *latitude*, *logitude*, dan *speed*. GPS U-Blox M8N digunakan memiliki tingkat *error* sekitar 2 meter. Selain itu, kondisi cuaca yang tidak bagus juga mempengaruhi keakuratan data dari GPS. Kondisi cuaca mendukung adalah kondisi cuaca yang paling mempengaruhi keakuratan data GPS. Serta, angin yang kencang juga dapat mempengaruhi data yang diterima GPS. GPS membutuhkan suplai daya 3.3 hingga 5 volt. Antarmuka GPS dengan Arduino ialah melalui pin komunikasi, pada tugas akhir ini melalui pin TX1 dan RX1. Rangkaian skematik GPS dan antarmukanya dengan Arduino Mega 2560 ditunjukkan oleh gambar 3.15 dan gambar 3.16.



Gambar 3.16. Antarmuka GPS dengan Arduino Mega 2560

3.2.12. Kompas



Gambar 3.17. Skematik Kompas HMC5883L [48].

Kompas yang digunakan adalah kompas digital HMC5883L, dimana keluaran dari kompas ini dimasukkan ke Arduino untuk diolah. Kompas memiliki rentang sudut dari 0 hingga 360 derajat. Antarmuka dengan Arduino ialah melalui pin I2C, dalam tugas akhir ini menggunakan pin SDA dan pin SCL atau pin 20 dan pin 21.

Tegangan kerja kompas menggunakan tegangan 3.3 volt yang disuplai oleh Arduino secara langsung. Rangkaian skematik kompas HMC5883L ditunjukkan oleh gambar 3.17, sedangkan antarmuka dengan Arduino ditunjukkan oleh gambar 3.18.



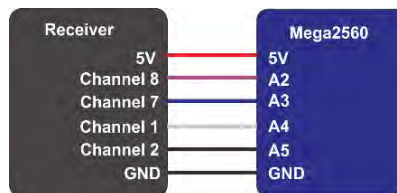
Gambar 3.18. Antarmuka Kompas HMC5883L dengan Arduino Mega 2560.

3.2.13. *Remote Control dan Receiver*



Gambar 3.19. Tombol-tombol Remot Kontrol untuk Kendali Buoyant Boat.

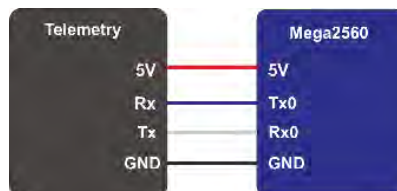
Buoyant Boat dikendalikan secara *wireless* dengan menggunakan sepasang remot kontrol Radiolink AT9 dan *receiver* R9D. Remot kontrol Radiolink AT9 sangat mudah digunakan. Pada gambar 3.19, ditunjukkan remot kontrol Radiolink AT9 beserta dengan keterangan tombol-tombol yang digunakan pada tugas akhir ini. Jika remot kontrol ada di *ground station*, lain halnya dengan *receiver*. *Receiver* terhubung dengan Arduino. Antarmuka antara *receiver* dengan Arduino ditunjukkan oleh gambar 3.20. Terdapat 4 *channel* yang digunakan, hal sesuai dengan keperluan untuk kendali manual (*throttle*, *aille*) dan pemilihan mode, serta *lock location* pada mode loiter. *Throttle* dan *aille* masing-masing adalah *channel* 1 dan *channel* 2 pada remot. *Channel* 8 (SwG) dengan tiga posisi untuk memilih mode dan *channel* 7 (SwE) dengan tiga posisi untuk mengunci lokasi yang diinginkan pada mode loiter.



Gambar 3.20. Antarmuka *Receiver* R9D dengan Arduino Mega 2560.

3.2.14. Telemetry

Telemetry yang digunakan pada tugas akhir ini adalah telemetry dengan frekuensi 433 MHz. Telemetry digunakan untuk komunikasi data secara *wireless* yang diperoleh pada *Buoyant Boat* untuk dikirim ke *ground station*. Antarmuka telemetry dengan Arduino ditunjukkan oleh gambar 3.21. Telemetry terhubung melalui pin komunikasi.



Gambar 3.21. Antarmuka Telemetry dengan Arduino Mega 2560.

3.2.15. *Driver Motor*

Driver motor yang digunakan adalah VN2SP30. Satu VN2SP30 untuk mengendalikan satu motor. Karena *Buoyant Boat* menggunakan dua motor, maka *driver motor* yang digunakan juga ada dua, untuk motor kiri dan motor kanan. Tegangan motor yang digunakan adalah 12 volt dari maksimal tegangan 15 volt. Pin-pin masukan pada *driver motor* ini terhubung dengan pin digital Arduino, seperti yang ditunjukkan pada gambar 3.22. Arus yang dapat dialirkan *driver motor* ini adalah hingga 30 Ampere.



Gambar 3.22. Antarmuka *Driver Motor* VN2SP30 dengan Arduino Mega 2560.

3.2.16. *Bilge Pump Motor*

Motor bilge pump pada *Buoyant Boat* terdapat dua buah, satu kiri dan satu kanan yang dapat dilihat pada gambar 3.23. Tegangan suplai motor sebesar 12 volt. *Bilge pump motor* ini adalah motor yang *waterproof*. Arah putaran motor ditentukan oleh pemberian tegangan DC pada kedua *input* motor yang dihasilkan dari *driver motor* VN2SP30.

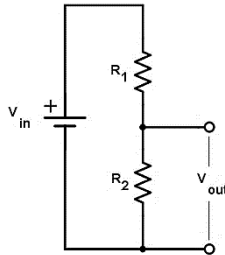


Gambar 3.23. *Bilge Pump Motor* pada *Buoyant Boat*.

3.2.17. *Analog to Digital Converter*

Analog to Digital Converter (ADC) digunakan untuk mengukur tegangan beban. Arduino memiliki pin ADC yang sangat mudah digunakan. Tegangan beban berada dalam rentang 12 volt, dimana tegangan ini tidak bisa langsung digunakan sebagai *input* pada Arduino. Maka dari itu, tegangan perlu diturunkan dengan sebuah

pembagi tegangan. Rangkaian pembagi tegangan dapat dilihat pada gambar 3.24. Tegangan keluaran kemudian dimasukkan dalam pin analog Arduino, dalam tugas akhir ini pada pin A9.



Gambar 3.24. Rangkain Pembagi Tegangan.

3.3. Perancangan Perangkat Lunak

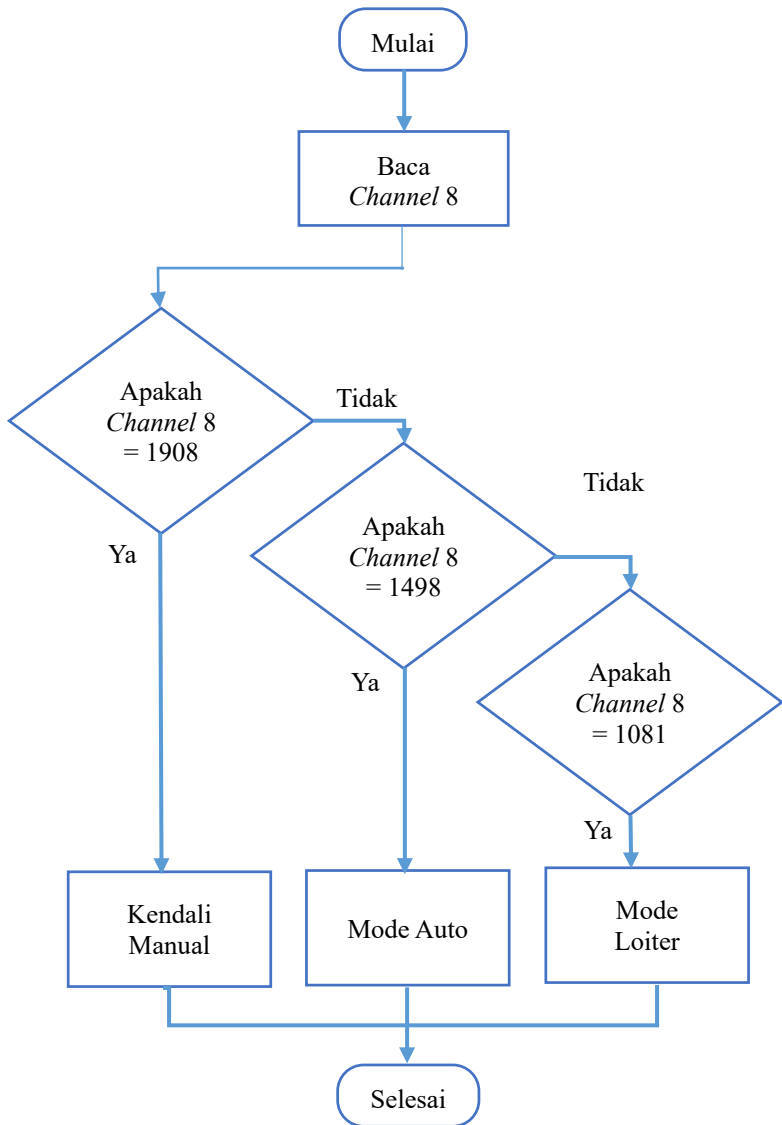
Perancangan perangkat lunak semuanya terintegrasi ke dalam Arduino Mega 2560. Algoritma pemrograman dirancang untuk melakukan beberapa proses yang dibagi menjadi tiga bagian utama, pertama yaitu algoritma kendali yang terdiri dari kendali manual, *auto*, dan mode loiter, kedua untuk pembacaan ADC, dan terakhir pemrograman antarmuka tampilan data pada monitor.

3.3.1. Perancangan Main Program (Lampiran A)

Main program dibuat untuk memilih mode kendali yang dikehendaki pengguna. Pemilihan mode dirancang sederhana agar pengguna dapat dengan mudah mengerti cara mengoperasikan *Buoyant Boat*. Cara memilih mode kendali cukup dengan menggeser posisi *channel* 8. *Channel* 8 terdapat tiga posisi, dimana pada program yang dibuat memiliki nilai maksimal 1908, nilai tengah 1498, dan nilai minimal 1081. Pembacaan *channel* 8 dengan menggunakan fungsi *pulseIn* yang ditunjukkan oleh penggalan program dibawah ini:

Ch8 = **pulseIn** (A2, **HIGH**, 35000);

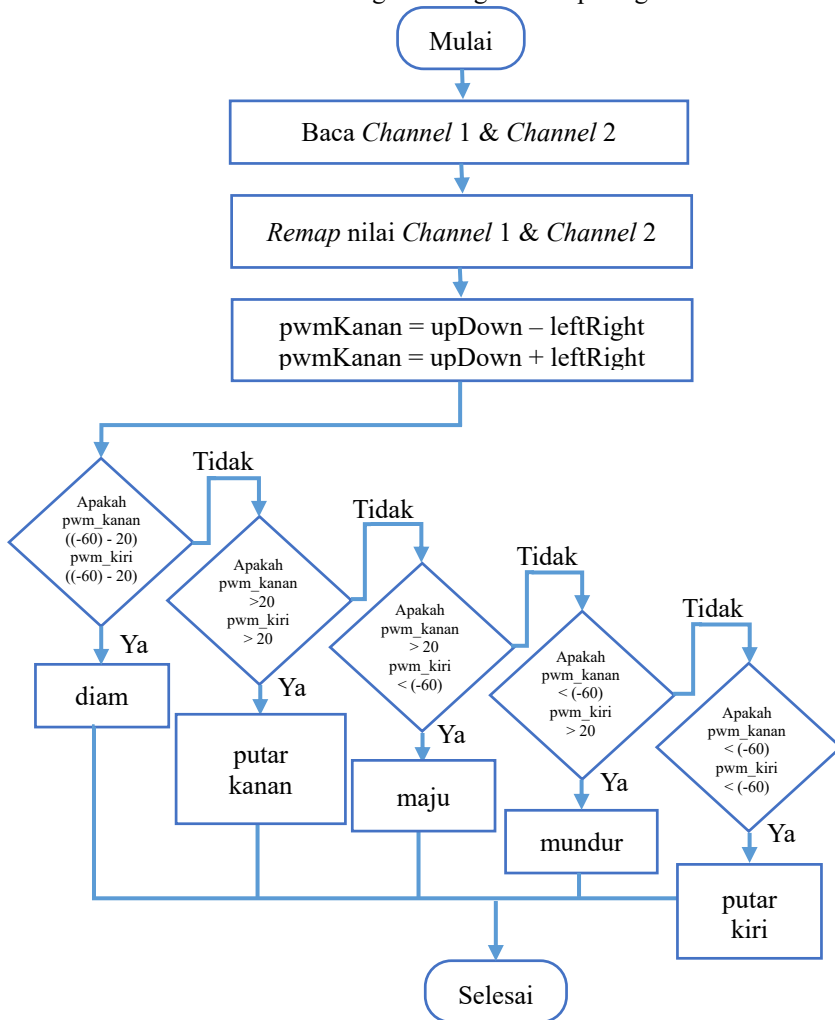
Pada program diatas, tipe pulsa yang dibaca adalah HIGH. *PulseIn()* menunggu pin dalam kondisi HIGH, kemudian mulai *timing*, dan menunggu kondisi pin LOW untuk menghentikan *timing*. Panjang pulsa dalam *microseconds* atau 0 jika tidak ada pulsa yang terbaca hingga batas waktu (*timeout*). Mode kendali ada tiga. Diagram alir pemilihan mode kendali ditunjukkan oleh gambar 3.25.



Gambar 3.25. Diagram Alir *Switching Mode*.

3.3.2. Perancangan Kendali Manual (Lampiran B)

Kendali manual mengikuti diagram alir pada gambar 3.26.



Gambar 3.26. Diagram Alir Kendali Manual.

Kendali manual dilakukan untuk menggerakkan *Buoyant Boat* melalui remot kontrol sesuai dengan perintah dari pengguna. Hal pertama yang dilakukan adalah memposisikan *channel 8* pada mode

kendali manual, yakni posisi tombol menuju kearah pengguna. Kemudian pengguna bisa menggerakkan *Buoyant Boat* dengan mudah dengan menggerakkan tombol *channel* 1 dan *channel* 2 yang merupakan tombol arah. Kode program untuk kendali manual yaitu sebagai berikut:

```
thro = pulseIn (A5, HIGH, 35000); //Channel 1
aile = pulseIn (A4, HIGH, 35000); //Channel 2
upDown = map(thro, THMIN, THMAX, -255, 255);
upDown = constrain (upDown, -255, 255);
leftRight = map(aile, AILMIN, AILMAX, -255, 255);
leftRight = constrain (leftRight, -255, 255);

pwm_kanan =(upDown-leftRight);
pwm_kiri = (upDown+leftRight);
pwm_kanan = constrain(pwm_kanan, -255, 255);
pwm_kiri = constrain(pwm_kiri, -255, 255);

if ((pwm_kanan>=-60 && pwm_kanan<=20)&&(pwm_kiri>=-
60 && pwm_kiri<=20)){
    //diam;
}
else if ((pwm_kanan>20)&&(pwm_kiri>20)){
    //Putar kanan;
}
else if ((pwm_kanan>20)&&(pwm_kiri<-60)){
    //Maju;
}
else if ((pwm_kanan<-60)&&(pwm_kiri>20)){
    //Mundur;
}
else if ((pwm_kanan<-60)&&(pwm_kiri<-60)){
    //Putar Kiri;
}
analogWrite(PWMA, abs(pwm_kanan));
analogWrite(PWMB, abs(pwm_kiri));
```

Fungsi *pulseIn()* digunakan untuk membaca pulsa yang diterima *receiver* dari remot.

Fungsi *map()* digunakan untuk me-*remap* nilai menjadi batas-

batas nilai yang diinginkan, disini memiliki batas dari -255 hingga 255. Nilai 255 merupakan nilai maksimal untuk PWM.

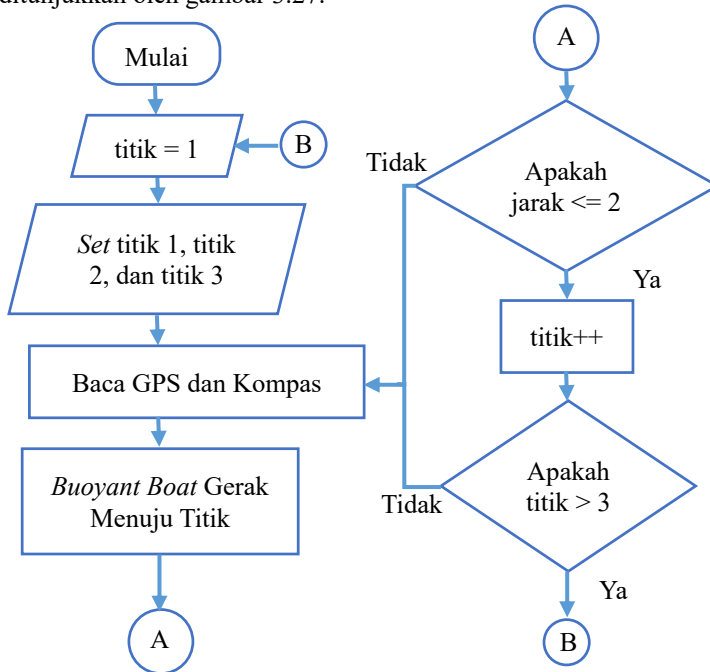
Fungsi *constrain()* digunakan untuk membatasi nilai agar nilai tidak berlebih ataupun kurang dari batas.

Fungsi *analogWrite()* digunakan untuk memberikan nilai analog pada pin PWM.

Fungsi *abs()* digunakan untuk membuat nilai menjadi nilai mutlak, karena PWM hanya bisa membaca nilai positif.

3.3.3. Perancangan Kendali *Auto* (Lampiran C)

Penggambaran program mode *auto* dengan diagram alir ditunjukkan oleh gambar 3.27.



Gambar 3.27. Diagram Alir Mode *Auto*.

Mode *auto* adalah fitur agar *Buoyant Boat* dapat beroperasi secara otomatis tanpa harus dikendalikan secara manual oleh pengguna. *Buoyant Boat* pada mode *auto* akan bergerak melintasi titik-titik yang telah ditetapkan sebelumnya di dalam program. Pada

tugas akhir ini, terdapat tiga titik koordinat yang dimasukkan kedalam program *auto*. GPS dan kompas mengatur navigasi untuk melalui titik-titik koordinat tersebut. Program untuk mengetahui jarak dan menghitung sudut *Buoyant Boat* dengan titik koordinat yang dituju diakses dari GPS dengan kode program seperti berikut:

```
distanceKmtitik =
    (unsigned long)TinyGPSPlus::distanceBetween(
        gps.location.lat(), gps.location.lng(),
        titik_lat, titik_lon);
courseTotitik =
    TinyGPSPlus::courseTo(
        gps.location.lat(), gps.location.lng(),
        titik_lat, titik_lon);
```

Jarak yang diperoleh dari program diatas berada dalam satuan meter. *Titik_lat* dan *titik_lon* adalah titik koordinat yang di set. *Titik_lat* adalah *latitude* dan *titik_lon* adalah *longitude*. Sudut *Buoyant Boat* terhadap titik yang dituju memiliki rentang nilai dari 0 hingga 359 derajat. Pertambahan nilai sudut tersebut sesuai arah jarum jam.

Sudut *Buoyant Boat* didapat dari nilai kompas. Dimana nilai sudut kompas juga memiliki rentang nilai dari 0 hingga 359 derajat. Metode untuk menggerakkan *Buoyant Boat* secara otomatis menuju titik yang dituju yaitu dengan membandingkan antara sudut yang didapat dari kompas dengan sudut dari GPS, serta dengan memanfaatkan nilai jarak *Buoyant Boat* ke titik tujuan.

Selanjutnya, ketika sudah berjarak 2 meter (jarak dapat disesuaikan, pada tugas akhir ini diatur berjarak 2 meter) dari titik tujuan pertama, maka titik tujuan berubah menjadi titik tujuan kedua. Begitu pula hingga titik ketiga. Ketika sudah lebih dari titik ketiga, maka titik kembali ke titik 1 kembali. Hal ini pada program ditulis sebagai berikut:

```
titik = 1;
switch (titik){
    case 1:
        titik_lat = -7.286343, titik_lon = 112.795952; //danau 8
        break;
    case 2:
        titik_lat = -7.286410, titik_lon = 112.796014; //danau 8
        break;
```

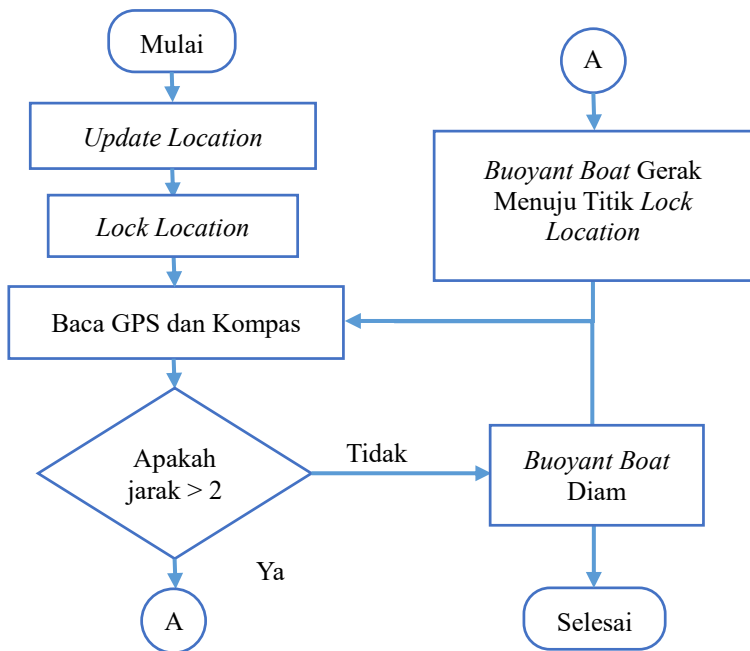
```

case 3:
    titik_lat = -7.286299, titik_lon = 112.796076; //danau 8
    break;
}
if (distanceKmtitik <= 2){
    if (++titik > 3) titik = 1;
}

```

3.3.4. Perancangan Mode *Loiter* (Lampiran D)

Diagram alir mode *loiter* ditunjukkan oleh gambar 3.28.



Gambar 3.28. Diagram Alir Mode *Loiter*.

Mode *loiter* merupakan fitur utama dari *Buoyant Boat*. Mode ini diperlukan saat ROV (saat *Buoyant Boat* dioperasikan bersama dengan ROV) akan menyelam. Dimana *Buoyant Boat* akan menjaga posisinya pada koordinat yang telah di *lock*. Pada tugas akhir ini, *Buoyant Boat* dapat menjaga daerah yang dikunci hingga radius 2

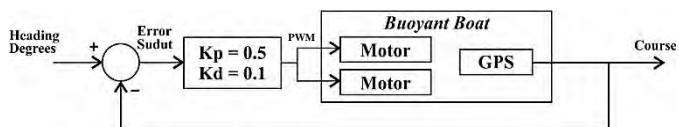
meter. Saat *Buoyant Boat* masih berada dalam radius 2 meter tersebut, *Buoyant Boat* diam. Jika *Buoyant Boat* terbawa arus dan berpindah sejauh lebih dari 2 meter, maka *Buoyant Boat* akan bergerak kembali menuju titik lokasi yang dikunci. Logika pergerakan *Buoyant Boat* mode *loiter* hampir sama dengan mode *auto*. Hanya saja, pada mode *auto* titik-titik koordinat di set melalui program, sedangkan titik lokasi yang dikunci pada mode *loiter* dilakukan secara langsung melalui remot kontrol.

Pada diagram alir gambar 3.28 diatas, *update location* merupakan perintah dari pengguna untuk menentukan titik lokasi yang ingin disalami dengan ROV. Tentunya menentukan titik lokasi dengan cara menggerakkan *Buoyant Boat* secara manual oleh pengguna. Jika sudah menemukan lokasi yang diinginkan, memindahkan posisi *channel 7* ke posisi untuk mengunci lokasi, yaitu posisi tengah (bernilai 1498 terbaca dengan fungsi *pulseIn()*). Dengan begitu, *Buoyant Boat* akan melakukan fungsi *loiter* seperti yang telah dijelaskan sebelumnya.

3.3.5. Perancangan Kontrol PID

Kontrol PID pada tugas akhir ini memiliki keluaran berupa sinyal PWM yang mempengaruhi letak *Buoyant Boat*. Sinyal PWM ini digunakan untuk mengontrol jumlah arus yang dihasilkan *driver* motor untuk menggerakkan motor dengan kecepatan yang diinginkan. Kontrol PID yang digunakan hanya P dan D saja, *proportional* dan *differential*, karena motor yang digunakan untuk menggerakkan *Buoyant Boat* pelan, namun masih kuat. Sehingga, ketika angin yang berhembus dapat menyebabkan *Buoyant Boat* berpindah.

Pada tugas akhir ini, digunakan kontrol PD, dengan gain P dan D sebesar $K_p = 0.5$ dan $K_d = 0.1$. Diagram blok kontrol PID yang digunakan ditunjukkan oleh gambar 3.29.



Gambar 3.29. Diagram Blok Kontrol PID.

Implementasi kontrol PID dilakukan di mikrokontroler Arduino Mega 2560. Algoritma dari kontrol PID yang dirancang untuk sistem ini meliputi pencarian nilai *error* dan dilanjutkan dengan pencarian

proportional error dan *derivative error*. Setelah nilai-nilai tersebut didapatkan, maka dengan perumusan PID akan didapatkan nilai keluaran PWM. Kode program untuk perhitungan PID pada tugas akhir ini yaitu seperti berikut ini:

```
bedaSudut = headingDegrees – courseTotitik;
if(bedaSudut < -180) bedaSudut += 360;
else if(bedaSudut > 180) bedaSudut -= 360;
proportionalSudut = 0.5*bedaSudut;
derivativeSudut = 0.1*(bedaSudut - previous_bedaSudut);
outputSudut = proportionalSudut + derivativeSudut;
previous_bedaSudut = bedaSudut;

throttle = 147;
steer = outputSudut;

kiri = throttle - steer;
kanan = throttle + steer;

kanan = constrain(kanan, -255, 255);
kiri = constrain(kiri, -255, 255);

analogWrite(PWMA, abs(kiri));
analogWrite(PWMB, abs(kanan));
```

Pada program diatas, beda sudut sebagai *error* sudut mula-mula dikonversi agar memiliki rentang nilai 0 hingga 180 derajat searah jarum jam, dan bernilai 0 hingga -180 derajat dengan arah berlawanan dengan arah jarum jam. Gain Kp dan Kd yang digunakan masing-masing bernilai 0.5 dan 0.1. Nilai *throttle* yang digunakan sebesar 147 agar nilai maksimum PWM kiri dan PWM kanan tidak melebihi 255, karena dapat membuat suatu *error* terhadap gerak motor. Meskipun begitu, tetap dilakukan antisipasi kelebihan nilai PWM dengan fungsi *constrain()*, sehingga nilai PWM maksimal adalah 255.

3.3.6. Perhitungan Kebutuhan Daya

Motor sebagai penggerak *Buoyant Boat* terdapat 2 buah dengan kebutuhan tegangan dan arus sebesar 12 Volt 3.3 A. Total daya yang diperlukan menjadi 79.2 Watt.

Panel surya yang digunakan memiliki kapasitas 100 Wp per

modul panel surya. Efisiensi panel surya berada di rentang 20 hingga 30%. Oleh karena itu, diperlukan minimal 3 buah panel surya. Jika efisiensi 30%, maka total daya yang dapat dihasilkan oleh 3 modul panel surya yaitu sebesar 90 Watt. Secara matematika, dituliskan:

$$\begin{aligned}
 \text{Kebutuhan daya motor} &= 12 \text{ V} \times 3.3 \text{ A} \times 2 \\
 &= \mathbf{79.2 \text{ Watt.}} \\
 \text{Daya yang dihasilkan PV} &= 30\% \times 100 \text{ Wp} \times 3 \\
 &= \mathbf{90 \text{ Watt.}} \\
 \text{Selisih daya} &= \text{Suplai} - \text{Kebutuhan} \\
 &= \text{Daya yang dihasilkan PV} - \\
 &\quad \text{Kebutuhan daya motor} \\
 &= 90 - 79,2 \\
 &= \mathbf{10.2 \text{ Watt.}}
 \end{aligned}$$

Dari hasil perhitungan, terdapa selisih daya sebesar 10.2 Watt. Selisih daya ini berguna sebagai suplai daya elektronik *Buoyant Boat*.

3.3.7. Pembacaan ADC (*Analog to Digital Converter*)

Pembacaan nilai ADC disini bertujuan untuk memonitor tegangan beban. Arduino sudah memiliki ADC yang mudah untuk digunakan. Kode program pembacaan nilai ADC tegangan beban yaitu seperti berikut (lampiran E):

```
load_volt = (5.0 * analogRead(A10) / 980.0) * 10;
```

Pada program ini, menggunakan fungsi *analogRead()* yang membaca nilai pada pin A10.

3.3.8. Antarmuka Data Pada Monitor dengan PLX-DAQ v2.10 (Lampiran E)

Antarmuka tampilan data menggunakan *software excel* PLX-DAQ 2.10 (*open source*) yang mudah dalam penggunaannya. *Software* ini merupakan pengembangan dari *microsoft excel* yang mana dapat digunakan oleh *developer* untuk menampilkan data serial. Dalam menampilkan data, penulisan kode program harus mengikuti format penulisan program yang telah ditetapkan oleh *developer* PLX-DAQ. Kode program pada Arduino untuk menghubungkan PLX-DAQ yaitu seperti berikut:

```
void setup() {
    Serial.println("CLEARDATA");
    Serial.println("LABEL,Computer Time,Time (Milli Sec.),
    Load Volt, Lattitude, Longitude, Speed, titik");
}
```

```

void loop(){
  for (i=0; i<=25000; i++){
    if(i==100){
      Serial.print("DATA,TIME,");
      Serial.print(milli_time);
      Serial.print(",");
      Serial.print(load_volt);
      Serial.print(",");
      Serial.print(recent_lat, 6);
      Serial.print(",");
      Serial.print(recent_lon, 6);
      Serial.print(",");
      Serial.println(recent_speed, 2);
    }
  }
}

```

Maka pada excel PLX-DAQ 2.10 data yang muncul adalah seperti yang ditunjukkan pada gambar 3.29.

The screenshot shows the PLX-DAQ v2.10 software interface. On the left, a table displays data collected from a sensor. The table has six columns: Computer Time, Time (Milli Sec.), Load Volt, Latitude, Longitude, and Speed. The data rows show a sequence of measurements over time, with the Load Volt column showing values like 32, 93, 153, 214, 273, 333, 394, 453, 514, 574, 633, 694, 753, 814, 874, 933, 994, 1054, 1114, 1174, 1234, 1294, 1354, 1415, 1474, 1534, and 1595. On the right, the 'Control' panel is visible, showing settings for the PLX-DAQ interface, including checkboxes for 'Custom Checkboxes 1', 'Custom Checkboxes 2', and 'Custom Checkboxes 3', a 'Port' field set to 10, a 'Reset on Connect' button, a 'Reset Timer' button, a 'Clear Columns' button, and a 'Display direct debug' checkbox. The 'Sheet name to post to' field is set to 'Further sheet'.

	A	B	C	D	E	F
	Computer Time	Time (Milli Sec.)	Load Volt	Latitude	Longitude	Speed
2	11.08.19 AM	32	0	0	0	0
3	11.08.19 AM	93	0	0	0	0
4	11.08.19 AM	153	0	0	0	0
5	11.08.19 AM	214	0	0	0	0
6	11.08.19 AM	273	0	0	0	0
7	11.08.19 AM	333	0	0	0	0
8	11.08.19 AM	394	0	0	0	0
9	11.08.19 AM	453	0	0	0	0
10	11.08.20 AM	514	0	0	0	0
11	11.08.20 AM	574	0	0	0	0
12	11.08.20 AM	633	0	0	0	0
13	11.08.20 AM	694	0	0	0	0
14	11.08.20 AM	753	0	0	0	0
15	11.08.20 AM	814	0	0	0	0
16	11.08.20 AM	874	0	0	0	0
17	11.08.20 AM	933	0	0	0	0
18	11.08.20 AM	994	0	0	0	0
19	11.08.20 AM	1054	0	0	0	0
20	11.08.20 AM	1114	0	0	0	0
21	11.08.20 AM	1174	0	0	0	0
22	11.08.20 AM	1234	0	0	0	0
23	11.08.20 AM	1294	0	0	0	0
24	11.08.20 AM	1354	0	0	0	0
25	11.08.20 AM	1415	0	0	0	0
26	11.08.20 AM	1474	0	0	0	0
27	11.08.20 AM	1534	0	0	0	0
28	11.08.20 AM	1595	0	0	0	0

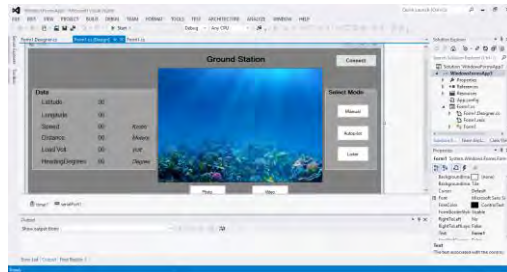
Gambar 3.30. PLX-DAQ v2.10.

PLX-DAQ 2.10 ini memiliki kekurangan, yakni secara tiba-tiba *not responding* jika dalam proses akuisisi data atau *log data window excel* diubah-ubah.

3.3.9. Antarmuka Data Pada Monitor

Antarmuka tampilan data menggunakan Visual Studio 2012 dirancang agar tampilan antarmuka menjadi lebih menarik. Berikut merupakan tampilan saat perancangan GUI dengan *windows form*

menggunakan Visual Studio 2012.



Gambar 3.31. Antarmuka Monitoring Data.

Untuk menampilkan data yang akan di monitor secara serial menggunakan program dibawah ini.

```
string serialku;
int cnt_display = 0;
private void timer1_Tick(object sender, EventArgs e){
    serialku = serialPort1.ReadLine();
    cnt_display++;
    if (cnt_display > 50){
        if (serialku[0] == 'a'){
            label2.Text = serialku.Remove(0, 1);
        }
        else if (serialku[0] == 'b'){
            label6.Text = serialku.Remove(0, 1);
        }
        else if (serialku[0] == 'c'){
            label7.Text = serialku.Remove(0, 1);
        }
        else if (serialku[0] == 'd'){
            label8.Text = serialku.Remove(0, 1);
        }
        else if (serialku[0] == 'e'){
            label11.Text = serialku.Remove(0, 1);
        }
        else{
            label12.Text = serialku.Remove(0, 1);
        }
        cnt_display = 0;
    }
}
```

.....*Halaman ini sengaja dikosongkan*.....

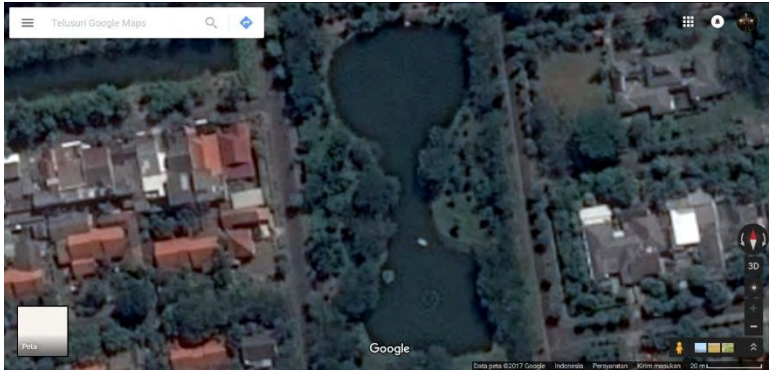
BAB IV PENGUJIAN DAN ANALISIS

Bab ini membahas tentang pengujian dari sistem yang telah dirancang beserta analisis data yang diperoleh saat pengujian. Pengujian-pengujian yang dilakukan meliputi pengujian GPS, pengujian kompas digital, pengujian kendali yang terdiri dari kendali manual, mode *auto*, dan mode *loiter*, pengujian daya yang dihasilkan panel surya, dan pengujian *graphical user interface* (GUI). *Buoyant Boat* yang berhasil direalisasikan ditunjukkan pada gambar 4.1.



Gambar 4.1. *Buoyant Boat* Tampak Belakang.

Pengujian *Buoyant Boat* secara real dilakukan di Danau Delapan Institut Teknologi Sepuluh Nopember (ITS), Surabaya. Tempat ini berlokasi di jalan Teknik Mesin, Keputih, Sukolilo, Kota Surabaya, Jawa Timur – 60111. Danau 8 ITS dilihat dengan google maps diperlihatkan oleh gambar 4.2.



Gambar 4.2. Danau 8 ITS.
(Sumber: Google Maps, 2017)

4.1. Pengujian GPS

Sebelum GPS digunakan dalam *Buoyant Boat*, GPS di tes terlebih dahulu agar dicapai komunikasi data yang terbaik. Pada pengujian yang dilakukan didapat *baudrate* GPS yang terbaik pada 9600. Sehingga data yang ditampilkan dari GPS dapat kontinyu. Jika *baudrate* GPS tidak pada 9600, data dari GPS tidak kontinyu dan terkadang berhenti. Berikut data yang dihasilkan GPS pada *baudrate* 9600 yang ditunjukkan pada tabel 4.1.

Tabel 4.1. Data GPS.

<i>Sat</i>	<i>Latitude</i>	<i>Longitude</i>	<i>Course</i>	<i>Speed</i>	<i>Distance</i>
5	-7.283934	112.796524	322.20	0.65	12219272
6	-7.283936	112.796524	322.20	0.63	12219272
6	-7.283938	112.796524	322.20	0.63	12219272
6	-7.283939	112.796524	322.20	0.76	12219272
7	-7.283939	112.796524	23.35	0.85	55201603
7	-7.283936	112.796524	23.35	0.67	55201603
7	-7.283936	112.796524	23.35	0.61	55201603

GPS diuji coba di atap gedung A206. Data yang diakases dari GPS ada beberapa, yaitu jumlah *satellite*, *HDOP* (*Horizontal Dilution of Precision*), *Latitude*, *Longitude*, *Date*, *Time*, *Altitude*, *Course*, *Speed*, *Card*, dan *Distance*. Namun, pada tugas akhir ini yang digunakan hanya *latitude*, *longitude*, *course*, *distance*, dan

speed. Penentuan titik koordinat dengan GPS memiliki *error* sekitar 2 meter. Gangguan juga dapat disebabkan oleh angin yang sedikit kencang dan kondisi cuaca yang berawan. Saat angin sedikit kencang, data koordinat (*latitude dan longitude*) dapat bergeser. Pergeserannya sekitar 2 meter dari titik koordinat sebenarnya.

Sudut *course* memiliki rentang nilai dari 0 hingga 360 searah jarum jam. Pemetaannya seperti pada gambar 4.3 berikut. Saat pengujian *course* ini yang dijadikan acuan adalah Jepang dan London. Sudut *course* GPS dari Indonesia (tepatnya laboratorium A206) ke London bernilai 322 derajat, sedangkan dari Indonesia ke Jepang bernilai 23 derajat.



Gambar 4.3. Pemetaan Sudut GPS.

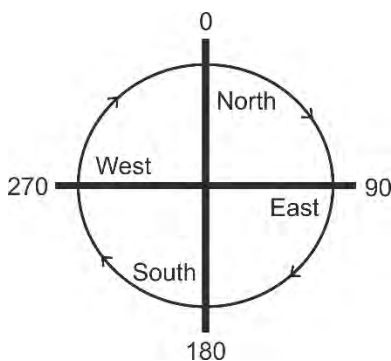
4.2. Pengujian Kompas Digital

Komunikasi kompas digital HMC5883L ialah melalui pin I2C pada arduino. Data sudut dari kompas yang terbaca yaitu 0 derajat untuk arah utara dan berputar searah jarum jam dengan nilai maksimal 359.9. Atau nilai kompas diantara $0 < \text{sudut} < 360$. Kompas sangat dipengaruhi oleh medan magnet, oleh karena itu, peletakan kompas harus terhindar dari pengaruh medan magnet. Data sudut dari kompas disajikan dalam tabel 4.2 berikut. Pemetaan sudut ditunjukkan oleh gambar 4.4.

Tabel 4.2. Data Sudut Kompas.

0.05	27.90	57.73	103.00	200.57	300.61	338.10
1.10	31.84	60.16	120.09	204.01	301.71	338.99
1.93	35.34	63.60	128.02	207.23	303.03	339.00

2.70	39.47	64.99	134.47	213.32	304.45	339.25
3.70	42.98	66.18	141.60	221.08	305.94	340.07
5.57	45.16	67.09	155.62	228.50	306.95	341.53
6.15	46.49	69.76	166.93	238.29	307.19	342.52
7.79	47.36	70.68	173.17	241.96	309.17	343.12
9.55	47.68	74.42	177.52	250.69	311.94	343.89
10.84	48.78	77.15	178.82	256.20	315.31	344.60
14.70	51.18	80.27	180.06	260.85	317.57	346.52
16.40	52.71	83.62	184.55	269.68	321.28	349.68
18.69	53.94	87.14	187.84	271.02	323.46	351.63
21.35	54.58	89.80	192.28	275.49	330.99	356.70
24.79	55.46	93.80	195.78	281.66	336.79	358.43
27.52	55.93	99.05	197.79	291.73	337.88	359.05



Gambar 4.4. Pemetaan Sudut Kompas.

4.3. Pengujian Kendali

Pengujian kendali dilakukan dengan menggerakkan remot kontrol sesuai SOP (*standard operating procedure*) untuk mengoperasikan *Buoyant Boat* (secara lengkap pada lampiran F). Terdapat tiga mode sistem kendali yang dirancang untuk mendukung ROV nantinya. Mode kendali tersebut yaitu manual, *auto*, dan *loiter*. Pemilihan mode kendali dengan main program (lihat lampiran A) yang telah dibuat dapat berfungsi dengan baik. Kecepatan maksimum *Buoyant Boat* yaitu sebesar 1.5 kmph diukur menggunakan GPS. *Buoyant Boat* masih dapat dipengaruhi oleh angin. Kemampuan GPS

dalam menerima data koordinat di tempat pengujian sangat baik. Indikasinya yaitu satelit yang mampu dideteksi sebanyak 12 satelit dari minimal 5 satelit untuk bisa mendapatkan titik *latitude* dan *longitude* suatu lokasi. Hal ini karena lokasi pengujian merupakan tempat terbuka dan saat pengujian kondisi langit sangat cerah.

4.3.1. Pengujian Kendali Manual

Sistem kendali manual bekerja dengan baik. Pengujian kendali manual sangat mudah. Hanya perlu menggerakkan *throttle* dan *aile* atau menggerakkan tombol arah. Kecepatan maksimal saat berada pada kendali manual sebesar 1.5 kmph dengan nilai PWM motor kanan dan PWM motor kiri sebesar 255. Kecepatan maksimal dicapai saat gerak maju. Gerak *Buoyant Boat* dengan kendali manual adalah kesegala arah, yaitu meliputi maju, mundur, kiri, kanan, maju serong kiri, maju serong kanan, mundur serong kiri, dan mundur serong kanan. Sistem kendali manual telah dilakukan beberapa kali pengujian di tempat berbeda, yaitu di kolam teknik fisika (gambar 4.5) dan di danau 8 ITS dengan menggunakan 1 PV (gambar 4.6) dan 3 PV (gambar 4.7).



Gambar 4.5. Pengujian Kendali Manual di Kolam Teknik Fisika ITS.



Gambar 4.6. Pengujian Kendali Manual di Danau 8 ITS dengan 1 PV.



Gambar 4.7. Pengujian Kendali Manual di Danau 8 ITS dengan 3 PV.

Pengujian di kolam Teknik Fisika mengalami kesalahan teknis yaitu lepasnya *propeller*, sehingga kendali tidak bisa maksimal. Hasilnya pun, *Buoyant Boat* hanya bisa berputar, karena hanya satu motor yang mendorong. Namun, dengan hasil tersebut diketahui

bahwa remot kontrol dapat terhubung dengan baik ke mikrokontroler yang berada pada *Buoyant Boat*.

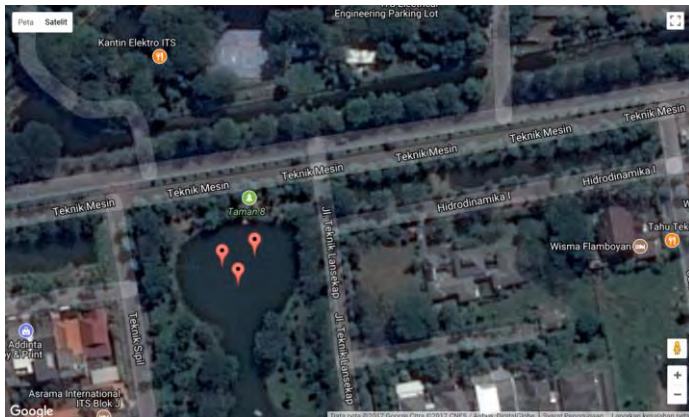
Pengujian di danau 8 ITS dengan 1 PV diketahui bahwa kendali manual bekerja dengan baik, namun gerakan terlihat tersendat-sendat ketika maju dengan kecepatan penuh. Setelah dianalisa, diketahui bahwa nilai PWM motor kanan dan motor kiri ada yang melebihi 255. Untuk memperbaikinya, maka pada pengujian selanjutnya nilai PWM dibatasi maksimal 255 ataupun -255. Teknik membatasi nilai PWM ini dengan memanfaatkan fungsi yang tersedia pada arduino, yakni fungsi *constrain()*. Hasilnya setelah dilakukan perbaikan program, *Buoyant Boat* dapat bergerak dengan maksimal tanpa tersendat-sendat. Pada pengujian dengan 3 PV pun sistem kendali manual dapat berfungsi dengan baik.

4.3.2. Pengujian Mode *Auto*

Pengujian sistem kendali mode *auto* dilakukan di Danau 8 ITS. Terdapat tiga titik koordinat yang di set untuk dilalui oleh *Buoyant Boat*. Titik-titik koordinat tersebut yaitu:

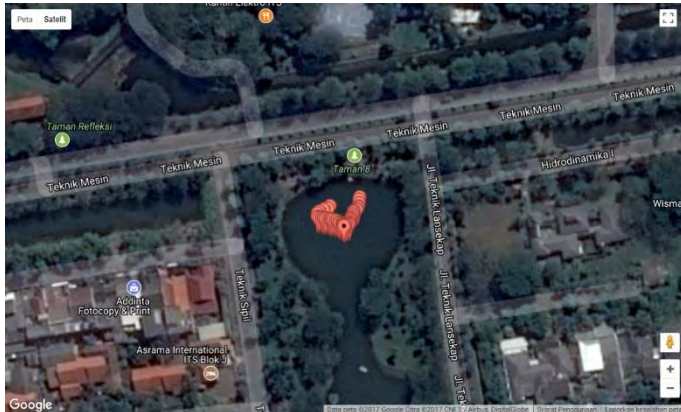
- (a) -7.286343,112.795952; (b) -7.286410,112.796014;
- (c) -7.286299,112.796076;

Plot titik (a) (b), dan (c) dengan *Google Maps* dapat dilihat pada gambar 4.8.



Gambar 4.8. Plot Titik Tujuan *Buoyant Boat*.

Target pencapaiannya ialah *Buoyant Boat* harus melalui ketiga titik tersebut dalam mode kendali *auto*. Setelah pengujian dilakukan, diketahui bahwa *Buoyant Boat* dapat melalui ketiga titik tersebut yang ditunjukkan oleh *trajectory Buoyant Boat* pada gambar 4.9, *trajcetory* dengan satelit dan gambar 4.10, *trajcetory* dengan peta.



Gambar 4.9. *Trajectory Buoyant Boat Satellite View.*



Gambar 4.10. *Trajectory Buoyant Boat Maps View.*

Dari gambar 4.10 terlihat bahwa titik-titik koordinat yang berhasil didata terlihat unik, dalam artian ada yang terlihat melenceng. Namun, masih dalam toleransi radius 2 meter, sehingga

Buoyant Boat masih tetap bisa menuju titik-titik koordinat tersebut. Hal ini karena saat pengujian juga terdapat angin, yang mana dapat mempengaruhi data titik koordinat yang diterima oleh GPS.

Buoyant Boat akan memindah titik koordinat tujuan ketika sudah berjarak 2 meter dengan titik koordinat tujuan yang pertama. Kecepatan maksimum *Buoyant Boat* saat berada dalam mode *auto* yaitu sebesar 1.37 kmph. Metode pergerakan *Buoyant Boat* yaitu *Buoyant Boat* akan mengarahkan diri menuju titik tujuan, sembari menggerakkan motor untuk menuju titik tujuan tersebut. Hal ini diperlihatkan oleh tabel 4.3 berikut.

Tabel 4.3. Data *Trajectory Buoyant Boat*.

Titik	<i>Latitude</i>	<i>Longitude</i>	<i>Distance</i>	<i>Speed</i>
1	-7.286333	112.795959	3	1.33
2	-7.286341	112.795944	2	1.37
2	-7.286348	112.795936	10	1.37
2	-7.286356	112.795921	10	1.22
2	-7.286362	112.795913	10	1.07
2	-7.286374	112.795913	9	1.09
2	-7.286375	112.795921	8	1.07
2	-7.286377	112.795928	9	1.00
2	-7.286378	112.795936	8	0.96
2	-7.286379	112.795936	8	1.06
2	-7.286380	112.795944	9	1.24
2	-7.286382	112.795944	9	1.2
2	-7.286382	112.795951	8	1.19
2	-7.286383	112.795959	8	1.26
2	-7.286384	112.795967	7	1.22
2	-7.286385	112.795967	7	1.2
2	-7.286386	112.795974	7	1.28
2	-7.286389	112.795982	6	1.26
2	-7.286393	112.795989	6	1.22
2	-7.286396	112.795997	5	1.19
2	-7.286398	112.795997	5	1.22
2	-7.286399	112.796005	4	1.3
2	-7.286400	112.796005	4	1.2
2	-7.286401	112.796012	4	1.26
2	-7.286402	112.796012	4	1.17

2	-7.286405	112.796012	3	1.24
2	-7.286406	112.79602	3	1.28
2	-7.286407	112.79602	3	1.31
2	-7.286408	112.79602	3	1.17
3	-7.286409	112.79602	2	1.07
3	-7.286406	112.796028	13	1.06
3	-7.286403	112.796033	13	1.02
3	-7.286402	112.796034	12	1.15
3	-7.286401	112.796035	12	1.02
3	-7.286395	112.796036	11	1.36
3	-7.286390	112.796038	10	1.35
3	-7.286384	112.796040	9	1.37
3	-7.286378	112.796042	8	1.13
3	-7.286371	112.796045	8	0.89
3	-7.286365	112.796049	8	0.93
3	-7.286359	112.796051	7	1.02
3	-7.286353	112.796053	6	1.02
3	-7.286348	112.796063	5	1.04
3	-7.286344	112.796062	5	1.11
3	-7.286339	112.796067	4	1.04
3	-7.286336	112.796069	3	1.02
3	-7.286324	112.796072	3	0.96
1	-7.286311	112.796074	2	0.96

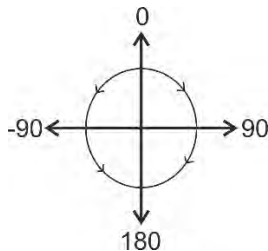
4.3.3. Pengujian Mode *Loiter*

Mode *loiter* dirancang untuk mendukung ROV saat akan mulai observasi di bawah air. Pengujian mode *loiter* ini dilakukan di Danau 8 ITS. Saat pengujian, kondisi cuaca cerah berawan. *Lock location* dengan remot kontrol dapat berfungsi dengan baik. Pada tugas akhir ini, tombol SwE (*channel 7*) pada remot yang digunakan untuk menyimpan titik koordinat atau *lock location*. Hasil yang diperoleh yaitu *Buoyant Boat* kembali ke posisi *lock location* ketika berpindah sejauh 2 meter dari titik *lock location*. Dokumentasi pengujian mode *loiter* ditunjukkan oleh gambar 4.11.



Gambar 4.11. Pengujian Mode *Loiter*.

Proses kembalinya *Buoyant Boat* adalah ketika *Buoyant Boat* terbawa arus sejauh lebih dari 2 meter. Logika berputar *Buoyant Boat* mengikuti nilai *error* sudut yang terbaca kompas dengan GPS. *Buoyant Boat* akan putar kiri atau putar kanan sesuai dengan arah ketika *Buoyant Boat* terbawa arus sejauh lebih dari 2 meter. Pemetaan nilai *error* sudut yang terbaca diperlihatkan oleh gambar 4.12.



Gambar 4.12. Pemetaan Nilai *Error* Sudut.

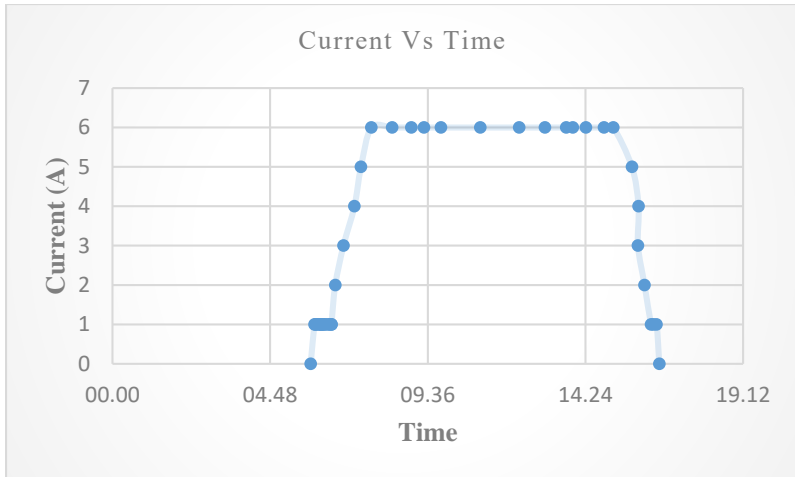
Metode mempertahankan posisi atau *lock location* sebenarnya ada 2, yaitu mode *loiter* ini sendiri dan *hold position*. Pada tugas akhir ini, dipilih mode *loiter* karena mode *loiter* lebih hemat energi dibandingkan dengan *hold position*. Hal ini karena *Buoyant Boat* hanya diam jika jarak dengan titik *lock location* masih berada dalam rentang 2 meter. Sedangkan, metode *hold position* langsung mempertahankan posisi dengan menggerakkan motor atau memberikan reaksi terhadap aksi dari arus air untuk mempertahankan posisi.

4.4. Pengujian Panel Surya

Pengujian panel surya disini dimaksudkan untuk mengetahui sejak kapan panel surya mampu memberikan suplai daya sebagai sumber energi baterai. Pengujian dilakukan di atas atap gedung A206. Terdapat 3 panel surya yang tersusun secara paralel pada tugas akhir ini, jadi total arus yang dihasilkan adalah hasil penjumlahan dari arus yang dihasilkan masing-masing panel surya. Panel surya diposisikan seperti posisi saat dipasang pada *Buoyant Boat* yaitu dengan kemiringan sekitar 0 derajat seperti yang terlihat pada gambar 4.13. Dari hasil pengujian, diketahui bahwa panel surya dapat menghasilkan arus mulai jam 06.09 hingga 16.34 WIB. Daya yang dihasilkan bergantung pada intensitas cahaya. Dalam grafik pada gambar 4.14 diketahui bahwa arus maksimal yang dapat dihasilkan panel surya adalah sebesar 6 A, yakni mulai jam 07.53 hingga 15.15 WIB. Tegangan yang dihasilkan panel surya ini sebesar 13.8 volt. Sehingga, potensi daya maksimum dari ketiga modul panel surya ini adalah 82.8 Watt. Pengukuran arus menggunakan MPPT *Solar Charge Controller* (T40). Selain itu, MPPT ini juga memiliki regulasi yang berguna untuk menjaga baterai Lipo agar tetap aman saat proses pengisian daya. Seperti yang diketahui secara umum bahwa baterai Lipo sangat rentan terbakar, meledak, dan sebagainya jika salah dalam memberikan perlakuan saat pengisian daya baterai. Pada pengujian ini, pengisian daya baterai berhenti jika tegangan baterai sebesar 12.6 volt dan mulai mengisi lagi jika tegangan baterai berada pada tegangan 10.5 volt.



Gambar 4.13. Pengujian 3 Modul Panel Surya.



Gambar 4.14. Grafik Arus Terhadap Waktu.

Saat pengujian, total daya yang dihasilkan 3 panel surya sebesar 82.8 Watt. Dengan begitu, efisiensi yang sebenarnya adalah 27.6% atau memiliki hasil yang mendekati hipotesa, yakni 30%. Pada perancangan diketahui bahwa kebutuhan daya dari 2 motor yang digunakan *Buoyant Boat* adalah sebesar 79.2 Watt. Oleh karena itu, surplus daya hanya sebesar 3.6 Watt. Berdasarkan hasil ini, maka penggunaan panel surya selanjutnya menyesuaikan kebutuhan daya yang diperlukan.

Saat pengujian *Buoyant Boat*, panel surya sangat bermanfaat untuk menyuplai daya baterai. Jadi, selama ada cahaya matahari, *Buoyant Boat* dapat beroperasi dengan bebas tanpa khawatir kehabisan daya baterai. Kalaupun daya baterai habis, yang diperlukan hanyalah menunggu hingga panel surya selesai mengisi daya baterai dan *Buoyant Boat* dapat dioperasikan kembali.

4.5. Monitoring Data

4.5.1. PLX-DAQ v2.10

Monitoring data pada *Buoyant Boat* ini adalah dengan menggunakan PLX-DAQ v2.10. Dimana ini adalah *microsoft excel* yang telah dikembangkan oleh seorang developer. Data ditransmisikan menggunakan sepasang telemetri, satu terhubung

dengan mikrokontroler melalui pin komunikasi RX0 TX0 dan satu lagi terhubung dengan laptop menggunakan USB. Hasil data yang diperoleh yaitu seperti pada gambar 4.15 dibawah ini. Transmisi data dengan PLX-DAQ v2.10 cukup stabil, diketahui dengan tidak adanya data yang *error*.

Meskipun cukup stabil, namun PLX-DAQ versi 2.10 ini memiliki kelemahan, yakni tidak boleh melakukan apapun pada *microsoft excel* yang telah dikembangkan ini saat melakukan *monitoring* data. Jika hal ini terjadi, maka *microsoft excel* secara tiba-tiba tidak akan merespon atau yang biasa diketahui dengan istilah *not responding*.

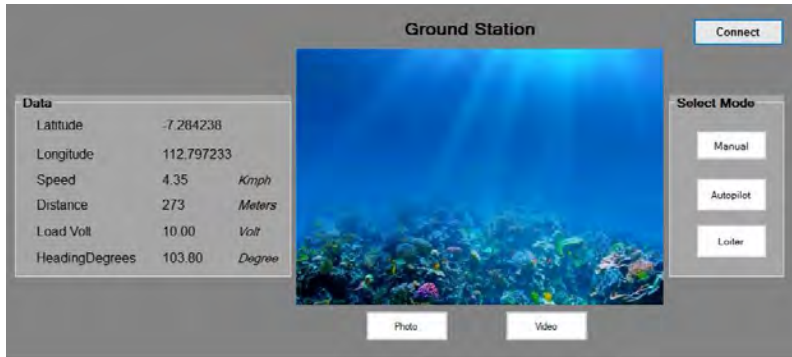
	A	B	C	D	E	F	G	H	I
1	Computer Time	Time (Milli Sec.)	Load Volt	Latitude	Longitude	PLX-DAQ for Excel "Version 2" by Net^Devil			
37	11.08.21 AM	2136	5.87	-7.286333	112.795959	<div> <div>PLX-DAQ v. 2.10</div> <div> <div>Control</div> <div> <input checked="" type="checkbox"/> Custom Checkbox 1 <input checked="" type="checkbox"/> Custom Checkbox 2 <input checked="" type="checkbox"/> Custom Checkbox 3 <input type="checkbox"/> Reset on Connect </div> <div> Reset Timer Clear Columns </div> </div> <div> <div>Settings</div> <div> Port: 10 Baud: 57600 </div> <div> Connect Pause logging </div> <div> Sheet name to post to: Further sheet Load (reload after renaming) </div> <div> Controller Messages: </div> <div> PLX-DAQ Status </div> <div> Do not move this window around while logging! That might crash Excel! </div> </div> </div>			
38	11.08.21 AM	2197	0	-7.286333	112.795959				
39	11.08.21 AM	2256	3.01	-7.286333	112.795959				
40	11.08.21 AM	2316	0	-7.286333	112.795959				
41	11.08.21 AM	2376	0	-7.286333	112.795959				
42	11.08.21 AM	2437	0	-7.286333	112.795959				
43	11.08.21 AM	2496	0	-7.286333	112.795959				
44	11.08.21 AM	2556	0	-7.286333	112.795959				
45	11.08.21 AM	2617	0	-7.286333	112.795959				
46	11.08.21 AM	2676	6.79	-7.286333	112.795959				
47	11.08.22 AM	2737	0	-7.286333	112.795959				
48	11.08.22 AM	2796	0	-7.286333	112.795959				
49	11.08.22 AM	2856	10	-7.286333	112.795959				
50	11.08.22 AM	2916	0	-7.286333	112.795959				
51	11.08.22 AM	2976	7.24	-7.286333	112.795959				
52	11.08.22 AM	3037	0	-7.286333	112.795959				
53	11.08.22 AM	3097	11.22	-7.286333	112.795959				
54	11.08.22 AM	3158	0	-7.286341	112.795944				
55	11.08.22 AM	3218	2.45	-7.286341	112.795944				
56	11.08.22 AM	3277	0	-7.286341	112.795944				
57	11.08.22 AM	3337	2.4	-7.286341	112.795944				
58	11.08.22 AM	3397	5.31	-7.286341	112.795944				
59	11.08.22 AM	3458	0	-7.286341	112.795944				
60	11.08.22 AM	3517	8.83	-7.286341	112.795944				
61	11.08.22 AM	3577	0	-7.286341	112.795944				
62	11.08.22 AM	3638	9.54	-7.286341	112.795944				
63	11.08.23 AM	3697	0	-7.286341	112.795944				

Gambar 4.15. Tampilan PLX-DAQ v2.10 saat *Monitoring Data*.

Nilai ADC terbaca pada kolom *Load Volt* (tegangan beban). Nilai tegangan yang terbaca adalah tegangan beban, sehingga berubah-ubah sesuai yang dibutuhkan oleh beban. Data titik koordinat terlihat pada kolom *Latitude* dan *Longitude*. Pada gambar 4.15 diatas juga terdapat data waktu yang menunjukkan waktu *monitoring data*.

4.5.2. Graphical User Interface (GUI)

Setelah sebelumnya memonitor data menggunakan PLX-DAQ v2.10, kali ini menggunakan GUI yang dibuat sendiri menggunakan Visual Studio 2012. GUI yang dibuat ditunjukkan oleh gambar 4.16.



Gambar 4.16. GUI untuk *Monitoring Data*.

Pada GUI diatas, terdapat terdapat beberapa data yang di monitor, antara lain: *latitude*, *longitude*, *speed*, *distance*, *load volt*, dan *headingDegrees*. *Latitude* dan *longitude* digunakan untuk mengetahui posisi *Buoyant Boat*. *Speed* untuk mengukur kecepatan *Buoyant Boat*, *distance* untuk mengetahui jarak antara *Buoyant Boat* dengan titik tujuan pada mode *loiter* dan *auto*. *Load Volt* untuk mengetahui tegangan beban yang digunakan, dan *headingDegrees* untuk mengetahui arah menghadap *Buoyant Boat*.

4.6. Analisis Keseluruhan Perancangan Sistem

Buoyant Boat untuk membantu komunikasi antara ROV dengan *ground station* dilengkapi dengan fitur kendali dan pemanfaatan panel surya. Pada *ground station* terdapat *display* data dan video. Data untuk memonitoring kondisi dari *Buoyant Boat* ataupun dari ROV nantinya. Sedangkan, *display* video untuk menampilkan video yang diambil oleh *Buoyant Boat* dan ROV, sehingga nantinya kamera terdapat dua buah, 1 untuk *Buoyant Boat* dan 1 untuk ROV. Kamera berguna saat mengendalikan *Buoyant Boat* maupun ROV.

Secara keseluruhan, *Buoyant Boat* mampu bekerja dengan baik sesuai tujuan pembuatan. Pertama, *Buoyant Boat* dapat membantu komunikasi antara ROV dengan *ground station* dengan baik. Kedua, daya yang diperlukan mampu disuplai oleh panel surya. Ketiga, fitur

lock location juga bekerja dengan baik sesuai perancangan.

GPS di tempat pengujian mampu mendeteksi 12 satelit, sehingga data yang diterima GPS sangat baik. Kompas yang digunakan pada *Buoyant Boat* juga bekerja dengan baik. Keandalan GPS dan kompas ini membuat fitur navigasi pada *Buoyant Boat* dapat bekerja dengan baik. Oleh karena itu, kendali *auto* dan *loiter* berhasil dilakukan. Kecepatan maksimal mode *auto* yaitu sebesar 1.37 Kmph. Sedangkan, fitur kendali lainnya, fitur kendali manual, tidak dipengaruhi oleh GPS dan kompas. Hanya perlu mengendalikan *Buoyant Boat* melalui *remote* sesuai keinginan dan memperhatikan *display monitor* (video) ketika *Buoyant Boat* sudah berada jauh dari jangkauan. Kecepatan maksimal saat pada mode manual yaitu sebesar 1.5 Kmph. Berdasarkan hasil tersebut diatas, diketahui bahwa komunikasi *remote* dan *receiver* berfungsi dengan baik.

Pemanfaatan panel surya sebagai suplai daya utama sangat penting digunakan. Hal ini karena, *Buoyant Boat* dirancang untuk beroperasi dalam jangka waktu yang relatif lama. Efisiensi panel surya yang digunakan adalah sebesar 27,6% sehingga daya maksimal yang dihasilkan hanya sebesar 82,8 W dari 300 W daya maksimal 3 panel surya (@100 Wp). Dengan daya 82,8 W tersebut mampu untuk menyuplai kebutuhan daya *Buoyant Boat* yang paling banyak diserap oleh 2 motor yang digunakan, yakni sebesar 79,2 W.

Kelengkapan di *ground station*, yakni *monitoring data* dan video mampu bekerja dengan baik. Dimana data-data yang diperoleh dari *Buoyant Boat* bisa diamati di *ground station* melalui GUI yang telah dibuat. Tampilan video juga bekerja dengan baik.

Hasil pengujian fitur-fitur yang dirancang pada tugas akhir ini, ditunjukkan pada tabel 4.4.

Tabel 4.4. Pengujian Fitur Overall.

<i>Buoyant Boat</i>	Kendali manual	<input checked="" type="checkbox"/>
	Mode <i>auto</i>	<input checked="" type="checkbox"/>
	Mode <i>loiter</i>	<input checked="" type="checkbox"/>
	Penggunaan panel surya	<input checked="" type="checkbox"/>
<i>Ground Station</i>	<i>Display Data</i>	<input checked="" type="checkbox"/>
	<i>Display Video</i>	<input checked="" type="checkbox"/>

BAB V

PENUTUP

5.1. Kesimpulan

Berdasarkan percobaan yang telah dilakukan pada pelaksanaan tugas akhir ini didapat beberapa kesimpulan sebagai berikut:

- a. *Buoyant Boat* merupakan inovasi baru yang berguna sebagai jembatan komunikasi antara ROV dengan *ground station* secara *wireless* tanpa adanya pengurangan sinyal yang melewati perbatasan udara dengan air.
- b. *Buoyant Boat* sebagai penghubung ROV dengan *ground station* menyediakan komunikasi berupa komunikasi kendali, navigasi, dan data (data dan video) secara *wireless*.
- c. *Buoyant Boat* memiliki fitur yang diperlukan untuk mendukung ROV melakukan kegiatan observasi, yaitu fitur kendali manual, *auto*, *loiter* (sebagai fitur utama), serta penggunaan panel surya untuk mendukung kebutuhan suplai daya, sehingga aktivitas observasi dapat berlangsung relatif lama.
- d. Sistem kendali *Buoyant Boat*, kendali manual, *auto*, maupun *loiter* dapat bekerja dengan baik. GPS dan kompas cukup handal digunakan sebagai perlengkapan navigasi *auto* dan *loiter*.
- e. Pada percobaan, diketahui bahwa panel surya dapat menghasilkan daya mulai dari pukul 06.09 hingga 16.34 WIB dengan efisiensi sebesar 27,6%.
- f. MPPT *Solar Charge Controller* memiliki regulasi yang membuat pengisian daya baterai Lipo menjadi aman.

5.2. Saran

Sebagai sarana pengembangan *Buoyant Boat* ini, maka terdapat beberapa saran dari penulis berdasarkan hasil yang diperoleh saat percobaan, yaitu sebagai berikut:

- a. Diperlukan adanya penelitian pada panel surya dengan kondisi panel surya bergerak, sehingga efisiensi panel surya yang digunakan pada *Buoyant Boat* bisa lebih baik.
- b. Pengembangan pada *Buoyant Boat* selanjutnya tidak hanya pada sistem, tetapi juga diperlukan pengembangan pada badal kapal, yakni dengan membuat badal kapal yang sesungguhnya, bukan lagi berupa perahu karet.

.....*Halaman ini sengaja dikosongkan*.....

DAFTAR PUSTAKA

- [1] PPPGL, "Pusat Penelitian dan Pengembangan Geologi Kelautan," 2016. [Online]. Available: www.mgi.esdm.go.id/content/morfologi-dasar-laut-indonesia. [Accessed September 2017].
- [2] F. Irawan and A. Yulianto, "Perancangn Prototype Robot Observasi Bawah Air dan Kontrol Hovering Menggunakan Metode PID Control," *Perancangn Prototype Robot Observasi Bawah Air dan Kontrol Hovering Menggunakan Metode PID Control*, pp. 63-70, 2015.
- [3] Indomiliter, "ROV Si Robot Penyelam Laut Dalam," 2015. [Online]. Available: <http://www.indomiliter.com/rov-si-robot-penyelam-laut-dalam/>. [Accessed September 2017].
- [4] S. Sivcev, J. A. D. Coleman, G. Dooly, E. Omerdic and D. Toal, "Closing The Gap between Industrial Robots and Underwater Manipulators," Washington, 2015.
- [5] J. Singh, D. Gandhi, M. Sanghani, P. Robi and S. Dwivedy, "Design and Development of Underwater Robot," in *International Conference on Robotics, Automation, Control and Embedded Systems - RACE 2015*, Chennai, 2015.
- [6] M. S. Gitakarma, K. U. Ariawan and N. A. Wigrha, "Alat Bantu Survey Bawah Air Menggunakan AMOBA, Robot Berbasis ROV," *Jurnal Sains dan Teknologi*, pp. 392-409, 2014.
- [7] L. Liu, S. Zhou and J.-H. Cui, "Prospects and Problems of Wireless Communication for Underwater Sensor Networks," University of Connecticut, Storrs, USA.
- [8] Assodivers, "Venom 3k-200," 2013. [Online]. Available: <http://www.assodivers.gr/en/equipment/rovs/venom/>. [Accessed 8 April 2017].
- [9] M. A. B. Yusof and S. Kabir, "Underwater Communication Systems: A Review," *Progress In Electromagnetics Research Symposium Proceedings, Marrakesh, Morocco*, pp. 803-807, 2011.
- [10] E. H. Henriksen, I. Schjolberg and T. B. Gjersvik, "UW MORSE: The UnderWater Modular Open Robot Simulation Engine,"

- Autonomous Underwater Vehicles (AUV) 2016*, vol. I, no. 16, pp. 261-267, 2016.
- [11] Mantech, "MANTECH ELECTRONICS: International Components Distributor," [Online]. Available: <http://www.mantech.co.za/datasheets/products/A000047.pdf>. [Accessed 06 12 2017].
 - [12] M. A. Jordan and L. B. Jorge, "Guidance of Underwater Vehicles With Cable Tug Perturbations Under Fixed and Adaptive Control Systems," *IEEE Journal of Oceanic Engineering*, vol. 33, no. 4, pp. 579-598, 2008.
 - [13] T. Shiino, K. Kawada and T. Yamamoto, "Gimbals control with the camera for aerial photography in RC helicopter," in *IEEE*, Seoul, South Korea , 2008.
 - [14] D. C. Meena and A. Devanshu, "Genetic Algorithm Tuned PID Controller for Process Control," in *International Conference on Inventive Systems and Control (ICISC-2017)*, Coimbatore, India , 2017.
 - [15] D. Purwanto, "Control Design Principles," Electrical Engineering, ITS, Surabaya.
 - [16] STMicroelectronics, "VNH2SP30-E (Automotive Fully Integrated H-Bridge Motor Driver," STMicroelectronics, Geneva, Switzerland, 2017.
 - [17] L. Kai-sheng, Z. Hui-hui and F. Ren-yuan, "Trends and foreground of overseas service robot's development," 2000.
 - [18] Pololu, "VNH2SP30 Motor Driver Carrier MD01B," Pololu Corporation, 2001. [Online]. Available: <https://www.pololu.com/product/706>.
 - [19] Rule, "Model 27S 1100 Gallons Per Hour," 2012.
 - [20] KingPumps, "Rule Pumps Automatic 12 Volt DC Bilge Pump, 1100 GPH 27S (D)," 2012. [Online]. Available: <http://store.waterpumpsupply.com/ruau12vodcbi1.html>. [Accessed 10 12 2017].
 - [21] F. T. Putri, M. Ariyanto, Ismoyo, M. Arozi, W. Caesarendra and M. R. Ibrahim, "Development of Unmanned Aerial Vehicle (UAV) Ornithopter with Wireless Radio Control," Semarang, 2016.

- [22] "72 MHz radios," The Balsa Workbench, 2017. [Online]. Available: http://www.balsaworkbench.com/?page_id=337. [Accessed 23 04 2017].
- [23] UnmannedTech, "100mW Ardupilot Unmanned Telemetry Kit V2 (433MHz)," 2017. [Online]. Available: <https://www.unmannedtechshop.co.uk/100mw-ardupilot-unmanned-telemetry-kit-v2-433mhz/>. [Accessed 10 12 2017].
- [24] 3DRobotics, "3DR RADIO V2 (Quick Start Guide)," 3D Robotics, New York, 2013.
- [25] W. Habibi, "Pembangunan Sistem Pelacakan dan Penelusuran Device Mobile Berbasis Global Positioning Sistem (GPS) Pada Platform Mobile Google Android," ITS, Surabaya, 2011.
- [26] UnmannedTech, "U-Blox Neo M8N with Compass," Unmanned Tech, 2016. [Online]. Available: <https://www.unmannedtechshop.co.uk/ublox-neo-m8n-gps-with-compass/>. [Accessed 10 12 2017].
- [27] U-Blox, "NEO-M8: U-Blox M8 Concurrent GNSS Modules," u-blox, South America.
- [28] B. Soen, "Kompas," Purwokerto.
- [29] ebay, "Gy-273 Hmc5883l Triple Axis Compass Magnetometer Sensor Module for Arduino 3v-5v," ebay.com, [Online]. Available: <https://www.ebay.com/p/Gy-273-Hmc5883l-Triple-Axis-Compass-Magnetometer-Sensor-Module-for-Arduino-3v-5v/567896943?iid=400362466832&trkparms=aid%3D222007%26algo%3DSIM.MBE%26ao%3D2%26asc%3D41376%26meid%3D6c7c26bddf5148debe5346b4f4377205%26pid%3D100005%26rk>. [Accessed 10 12 2017].
- [30] M. J. Caruso, "3-Axis Digital Compass IC HMC5883L," Honeywell International Inc, Plymouth, 2010.
- [31] B. Anto, E. Hamdani and R. Abdullah, "Portable Battery Charger Berbasis Sel Surya," *Jurnal Rekayasa ElektriKA*, vol. XI, no. 1, pp. 19-24, 2014.
- [32] xlsemi, "XL4005 Datasheet Preview," XLSEMI.COM, 2014.
- [33] Arfan, "5A DC-DC Volt Adjustable Step Down XL4005 Buck Converter Out 1.25V-32V," tokopedia.com, 2016. [Online]. Available: <https://www.tokopedia.com/arfanstore/5a-dc-dc-volt->

- adjustable-step-down-xl4005-buck-converter-out-125v-32v?trkid=f=Ca0000L000P0W0S0Sh00Co0Po0Fr0Cb0_src=search_page=1_ob=23_q=dc+dc+adjustable+buck+converter_catid=577_po=4. [Accessed 10 12 2017].
- [34] M. H. Assiddiqi, "Rancang Bangun Robot Pembersih Lantai Kolam Renang," Departemen Teknologi Elektro, ITS, Surabaya, 2017.
 - [35] "Introduction to Autodesk's AutoCAD 2015," [Online]. Available: <http://yukonstruct.com/event/introduction-to-autodesks-autocad-2015/>. [Accessed 1 10 2017].
 - [36] R. Shabrina, "Fungsi AutoCAD di Bidang Industri," DosenIT, 20 12 2016. [Online]. Available: <https://dosenit.com/software/fungsi-autocad>. [Accessed 1 10 2017].
 - [37] S. M. L. Kabir, I. Alam, M. R. Khan, M. S. Hossain, K. S. Rahman and N. Amin, "Solar Powered Ferry Boat for the Rural Area of Bangladesh," in *International Conference on Advances in Electrical, Electronic and System Engineering*, Putrajaya, Malaysia, 2016.
 - [38] J. Gutiérrez, F. Villa-Medina and M. Á. Porta-Gándara, "Autonomous Surface Vehicle for Measuring Water Body Parameters," in *Electronics, Robotics and Automotive Mechanics Conference*, Morelos, Mexico, 2010.
 - [39] M. Arzamendia, D. Gregor, D. G. Reina, S. L. Toral and R. Gregor, "Evolutionary Path Planning of an Autonomous Surface Vehicle for Water Quality Monitoring," in *9th International Conference on Developments in eSystems Engineering*, Liverpool, UK , 2016.
 - [40] M. Noorizadeh and N. Meskin, "Design of Small Autonomous Boat for Course-keeping Manuevers," in *4th International Conference on Control, Decision and Information Technologies (CoDIT'17)*, Barcelona, Spain, 2017.
 - [41] S. Siyang and T. Kerdcharoen, "Development of Unmanned Surface Vehicle for Smart Water Quality Inspector," in *13th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, Chiang Mai, Thailand, 2016.

- [42] S. Das, P. K. Sadhu and S. Chakraborty, "Green Sailing of Solar PV Powered Country Boat using Buck-Boost Chopper," in *International Conference on Circuit, Power and Computing Technologies [ICCPCT]*, Nagercoil, India, 2016.
- [43] R. Leiner, "Research Solar Power Boat – Data Management and Online Visualization," in *IEEE AFRICON*, Windhoek, South Africa, 2007.
- [44] K. Sakamoto, M. Kanaoka, H. Muraoka and R. Hojyo, "Soft-switching PWM Forward Power Converter with Auxiliary Active Clamped Capacitor for Solar Energy-Driven Boat," in *International Conference on Power Electronic Drives and Energy Systems for Industrial Growth*, Perth, WA, Australia, Australia , 1998.
- [45] StevannieToys, "PERAHU KARET BESTWAY HYDRO FORCE RAFT X2," tokopedia.com, 23 09 2015. [Online]. Available: https://www.tokopedia.com/stevannietoys/perahu-karet-bestway-hydro-force-raft-x2?trkid=f=Ca1496L000P0W0S0Sh00Co0Po0Fr0Cb0_src=directory_page=2_ob=24_q=_ref=csearch_catid=1496_po=33. [Accessed 14 12 2017].
- [46] Arduino, "Arduino MEGA 2560," [Online]. Available: https://www.arduino.cc/en/uploads/Main/arduino-mega2560_R3-sch.pdf. [Accessed 07 01 2018].
- [47] Posse, "Ublox M8N Breakout Not Detecting Satellites," ublox.com, 05 06 2017. [Online]. Available: <https://forum.ublox.com/index.php/11224/ublox-m8n-breakout-not-detecting-satellites>. [Accessed 15 12 2017].
- [48] L. Ada, "Adafruit HMC5883L Breakout - Triple-Axis Magnetometer Compass Sensor," Adafruit Industries, 2016.

.....*Halaman ini sengaja dikosongkan*.....

LAMPIRAN A

Main Program Buoyant Boat

```
#include "diffSteer.h"
#include "autopilot.h"
#include "loitermode.h"
#include "monitor.h"

int data;
int manual = 0;
int autopilot = 0;
int loiter = 0;

void setup(){
  pinMode(A2, INPUT); //Ch8
  pinMode(13, OUTPUT);

  setupdiffSteer();
  setupautopilot();
  setuploitermode();
  setupmonitor();
}

void loop(){
  loopmonitor();

  Ch8 = pulseIn (A2, HIGH, 35000);

  if(Ch8 > 1703){
    looppdiffSteer();
  }
  else if(Ch8 > 1290 && Ch8 < 1703){
    loopautopilot();
  }
  else if(Ch8 < 1290){
    looploitermode();
  }
  else{
  }
}
```

.....*Halaman ini sengaja dikosongkan*.....

LAMPIRAN B

Program Kendali Manual

```
#define THMAX 2075
#define THMIN 867
#define AILMAX 2068
#define AILMIN 911

#define IN1 6
#define IN2 7
#define IN3 4
#define IN4 5
#define PWMA 9
#define PWMB 10

float a =0.1;
float leftRight, upDown;
int thro, aile;
int pwm_kanan, pwm_kiri;

void setupdiffSteer(void) {
    pinMode(9, OUTPUT); // pwm a
    pinMode(10, OUTPUT); // pwm b
    pinMode(7, OUTPUT); //in 1
    pinMode(6, OUTPUT); //in 2
    pinMode(5, OUTPUT); //in 3
    pinMode(4, OUTPUT); //in 4
    pinMode(A4, INPUT); //kanan kiri //thro
    pinMode(A5, INPUT); //maju mundur //aile
    digitalWrite(A6, 1);
    digitalWrite(A5, 1);
    Serial.begin(57600);
}

void looppdiffSteer(void) {
    float distanceKmtitik = 25.12;
    thro = pulseIn (A5, HIGH, 35000);
    aile = pulseIn (A4, HIGH, 35000);
    if ((thro>(THMAX-THMIN)/2-100)&&(thro<(THMAX-THMIN)/2+100))
        thro=0;
    else if (thro<100) thro = (THMAX-THMIN)/2;
    if ((aile>(AILMAX-AILMIN)/2-100)&&(aile<(AILMAX-AILMIN)/2+100))
        aile=0;
```

```

else if (aile < 100) aile = (AILMAX-AILMIN)/2;
upDown = map(thro, THMIN, THMAX, -255, 255);
upDown = constrain (upDown, -255, 255);
leftRight = map(aile, AILMIN, AILMAX, -255, 255);
leftRight = constrain (leftRight, -255, 255);
  pwm_kanan =(upDown-leftRight);
  pwm_kiri = (upDown+leftRight);
  pwm_kanan = constrain(pwm_kanan, -255, 255);
  pwm_kiri = constrain(pwm_kiri, -255, 255);
  if ((pwm_kanan>=-60 && pwm_kanan<=20)&&(pwm_kiri>=-60 &&
pwm_kiri<=20)){
    digitalWrite(IN1, 0);
    digitalWrite(IN2, 0);
    digitalWrite(IN3, 0);
    digitalWrite(IN4, 0);
  }
  else if ((pwm_kanan>20)&&(pwm_kiri>20)){
    digitalWrite(IN1, 0);
    digitalWrite(IN2, 1);
    digitalWrite(IN3, 1);
    digitalWrite(IN4, 0);
  }
  else if ((pwm_kanan>20)&&(pwm_kiri<-60)){
    digitalWrite(IN1, 0);
    digitalWrite(IN2, 1);
    digitalWrite(IN3, 0);
    digitalWrite(IN4, 1);
  }
  else if ((pwm_kanan<-60)&&(pwm_kiri>20)){
    digitalWrite(IN1, 1);
    digitalWrite(IN2, 0);
    digitalWrite(IN3, 1);
    digitalWrite(IN4, 0);
  }
  else if ((pwm_kanan<-60)&&(pwm_kiri<-60)){
    digitalWrite(IN1, 1);
    digitalWrite(IN2, 0);
    digitalWrite(IN3, 0);
    digitalWrite(IN4, 1);
  }
  analogWrite(PWMA, abs(pwm_kanan));
  analogWrite(PWMB, abs(pwm_kiri));
}

```


LAMPIRAN C

Program Mode *Auto*

```
#define IN1 6
#define IN2 7
#define IN3 4
#define IN4 5
#define PWMA 9 //motor kiri
#define PWMB 10 //motor kanan

#include <TinyGPS++.h>
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_HMC5883_U.h>

static const uint32_t GPSBaud = 9600;
TinyGPSPlus gps;
Adafruit_HMC5883_Unified mag = Adafruit_HMC5883_Unified(12345);

float bedaSudut, previous_bedaSudut, proportionalSudut,
derivativeSudut, outputSudut, jarakProp;
float distanceKmtitik, previous_Distance, proportionalDistance,
derivativeDistance, outputDistance;
double recent_speed;

int titik = 1;
int kiri, kanan;

void setupautopilot(void) {
  pinMode(9, OUTPUT); // pwm a
  pinMode(10, OUTPUT); // pwm b
  pinMode(7, OUTPUT); //in 1
  pinMode(6, OUTPUT); //in 2
  pinMode(5, OUTPUT); //in 3
  pinMode(4, OUTPUT); //in 4
  Serial.begin(57600);
  Serial1.begin(GPSBaud);

  if (!mag.begin()) {
    Serial.println("Check your wiring!");
    while (1);
  }
}
```

```

static void smartDelay(unsigned long ms) {
    unsigned long start = millis();
    do {
        while (Serial1.available())
            gps.encode(Serial1.read());
    } while (millis() - start < ms);
}

static void printFloat(float val, bool valid, int len, int prec) {
    if (!valid) {
        while (len-- > 1)
            Serial.print('*');
        Serial.print(' ');
    }
    else {
        Serial.print(val, prec);
        int vi = abs((int)val);
        int flen = prec + (val < 0.0 ? 2 : 1); // . and -
        flen += vi >= 1000 ? 4 : vi >= 100 ? 3 : vi >= 10 ? 2 : 1;
        for (int i = flen; i < len; ++i)
            Serial.print(' ');
    }
    smartDelay(0);
}

static void printInt(unsigned long val, bool valid, int len) {
    char sz[32] = "*****";
    if (valid)
        sprintf(sz, "%ld", val);
    sz[len] = 0;
    for (int i = strlen(sz); i < len; ++i)
        sz[i] = ' ';
    if (len > 0)
        sz[len - 1] = ' ';
    Serial.print(sz);
    smartDelay(0);
}

void loopautopilot(void) {
    double courseTotitik;
    double titik_lat;
    double titik_lon;

```

```

int steer = 0;
int throttle = 0;

sensors_event_t event;
mag.getEvent(&event);

float heading = atan2(event.magnetic.y, event.magnetic.x);
float declinationAngle = 0.22;
heading += declinationAngle;

if (heading < 0)
    heading += 2 * PI;
if (heading > 2 * PI)
    heading -= 2 * PI;

float headingDegrees = heading * 180 / M_PI;

switch (titik){
    case 1:
        titik_lat = -7.286343, titik_lon = 112.795952; //danau 8
        break;
    case 2:
        titik_lat = -7.286410, titik_lon = 112.796014; //danau 8
        break;
    case 3:
        titik_lat = -7.286299, titik_lon = 112.796076; //danau 8
        break;
}

distanceKmtitik =
    (unsigned long)TinyGPSPlus::distanceBetween(
        gps.location.lat(), gps.location.lng(),
        titik_lat, titik_lon);
courseTotitik =
    TinyGPSPlus::courseTo(
        gps.location.lat(), gps.location.lng(),
        titik_lat, titik_lon);
smartDelay(0);

if (millis() > 5000 && gps.charsProcessed() < 10);

bedaSudut = headingDegrees - courseTotitik;
if(bedaSudut < -180) bedaSudut += 360;

```

```

else if(bedaSudut > 180) bedaSudut -= 360;
proportionalSudut = 0.5*bedaSudut;//0.055
derivativeSudut = 0.1*(bedaSudut - previous_bedaSudut);
outputSudut = proportionalSudut + derivativeSudut;
previous_bedaSudut = bedaSudut;

proportionalDistance = 50*distanceKmtitik;
derivativeDistance = 5*(distanceKmtitik - previous_Distance);
outputDistance = proportionalDistance + derivativeDistance;
previous_Distance = distanceKmtitik;

throttle = 147;           //175
steer = outputSudut;

kiri = throttle - steer;
kanan = throttle + steer;

kanan = constrain(kanan, -255, 255);
kiri = constrain(kiri, -255, 255);

if((bedaSudut >= -15 && bedaSudut <= 15)){ //maju
    digitalWrite(IN1, 0);
    digitalWrite(IN2, 1);
    digitalWrite(IN3, 0);
    digitalWrite(IN4, 1);
}
else if(bedaSudut > 15){ //belok kiri
    digitalWrite(IN1, 0); //
    digitalWrite(IN2, 0);
    digitalWrite(IN3, 0);
    digitalWrite(IN4, 1);
}
else if(bedaSudut < -15){ //belok kanan
    digitalWrite(IN1, 0);
    digitalWrite(IN2, 1);
    digitalWrite(IN3, 0); //
    digitalWrite(IN4, 0);
}
analogWrite(PWMA, abs(kiri));
analogWrite(PWMB, abs(kanan));
if(distanceKmtitik <= 3){
    if(++titik > 3) titik = 1;
}

```

LAMPIRAN D

Program Mode Loiter

```
#define CH8MAX 1908
#define CH8MID 1498
#define CH8MIN 1081

#define CH7MAX 1908 //1719 //Lock
#define CH7MID 1498 //1305 //Do after Lock
#define CH7MIN 1081

#define IN1 6
#define IN2 7
#define IN3 4
#define IN4 5
#define PWMA 9
#define PWMB 10

#include <TinyGPS++.h>
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_HMC5883_U.h>

double titik_lat = 0;
double titik_lon = 0;

int Ch8, Ch7;

void setuploitermode(void){
  pinMode(A2, INPUT); //Ch8
  pinMode(A3, INPUT); //Ch7

  Serial.begin(57600);
  Serial1.begin(GPSBaud);

  if(!mag.begin()){
    while (1);
  }
}

void looploitermode(void){
  Ch8 = pulseIn (A2, HIGH, 35000);
```

```

Ch7 = pulseIn (A3, HIGH, 35000);

double distanceKmtitik, courseTotitik;

int steer = 0;
int throttle = 0;

sensors_event_t event;
mag.getEvent(&event);

float heading = atan2(event.magnetic.y, event.magnetic.x);

float declinationAngle = 0.22;
heading += declinationAngle;

if (heading < 0)
    heading += 2 * PI;

if (heading > 2 * PI)
    heading -= 2 * PI;

float headingDegrees = heading * 180 / M_PI;

if(Ch7 > 1290 && Ch7 < 1703){ //Lock Location //tengah
    titik_lat = gps.location.lat();
    titik_lon = gps.location.lng();

    digitalWrite(IN1, 0);
    digitalWrite(IN2, 0);
    digitalWrite(IN3, 0);
    digitalWrite(IN4, 0);
    analogWrite(PWMA, 0); //kanan
    analogWrite(PWMB, 0); //kiri
}

else if((Ch7 < 1290)){ //paling bawah
    titik_lat;
    titik_lon;

    distanceKmtitik =
        (unsigned long)TinyGPSPlus::distanceBetween(
            gps.location.lat(),gps.location.lng(),
            titik_lat,titik_lon);

```

```

courseTotitik =
    TinyGPSPlus::courseTo(
        gps.location.lat(),gps.location.lng(),
        titik_lat,titik_lon);

smartDelay(0);

if (millis() > 5000 && gps.charsProcessed() < 10);

bedaSudut = headingDegrees - courseTotitik;
if(bedaSudut < -180) bedaSudut += 360;
else if(bedaSudut > 180) bedaSudut -= 360;
proportionalSudut = 0.5*bedaSudut;
derivativeSudut = 0.1*(bedaSudut);
outputSudut = proportionalSudut + derivativeSudut;

throttle = 147;
steer = outputSudut;

kiri = throttle - steer;
kanan = throttle + steer;

kanan = constrain(kanan, -255, 255);
kiri = constrain(kiri, -255, 255);

if(distanceKmtitik > 2){
    if((bedaSudut >= -15 && bedaSudut <= 15)){ //maju
        digitalWrite(IN1, 0);
        digitalWrite(IN2, 1);
        digitalWrite(IN3, 0);
        digitalWrite(IN4, 1);
    }
    else if(bedaSudut > 15){ //belok kiri
        digitalWrite(IN1, 1); //
        digitalWrite(IN2, 0);
        digitalWrite(IN3, 0);
        digitalWrite(IN4, 1);
    }
    else if(bedaSudut < -15){ //belok kanan
        digitalWrite(IN1, 0);
        digitalWrite(IN2, 1);
        digitalWrite(IN3, 1); //
        digitalWrite(IN4, 0);
    }
}

```

```

    }
    analogWrite(PWMA, abs(kiri));
    analogWrite(PWMB, abs(kanan));
}
else if(distanceKmtitik <= 2){
    digitalWrite(IN1, 0);
    digitalWrite(IN2, 0);
    digitalWrite(IN3, 0);
    digitalWrite(IN4, 0);
    analogWrite(PWMA, 0); //kanan
    analogWrite(PWMB, 0); //kiri
}
}
else{
    digitalWrite(IN1, 0);
    digitalWrite(IN2, 0);
    digitalWrite(IN3, 0);
    digitalWrite(IN4, 0);
    analogWrite(PWMA, 0); //kanan
    analogWrite(PWMB, 0); //kiri
}
}

```


LAMPIRAN E

Program *Monitoring Data*

```
#include <TinyGPS++.h>
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_HMC5883_U.h>

unsigned long int milli_time;
float load_volt;
//int data;
int i = 0;

void setupmonitor(void) {
    Serial.begin(57600);
    Serial1.begin(GPSBaud);

    if(!mag.begin()){
        while (1);
    }
}

void loopmonitor(void) {
    double titik_lat, titik_lon;
    double recent_lat, recent_lon, recent_speed;

    sensors_event_t event;
    mag.getEvent(&event);

    float heading = atan2(event.magnetic.y, event.magnetic.x);

    float declinationAngle = 0.22;
    heading += declinationAngle;

    if (heading < 0)
        heading += 2 * PI;

    if (heading > 2 * PI)
        heading -= 2 * PI;

    float headingDegrees = heading * 180 / M_PI;
```

```

recent_lat = gps.location.lat();
recent_lon = gps.location.lng();
recent_speed = gps.speed.kmph();

milli_time = millis();
load_volt = (5.0 * analogRead(A10) / 980.0) * 10; //harusnya dibagi 1023 jika
pembagi tegangannya pass

Serial.print("a");
Serial.println(recent_lat, 6);
Serial.print("b");
Serial.println(recent_lon, 6);
Serial.print("c");
Serial.println(recent_speed, 2);
Serial.print("d");
Serial.println(distanceKmtitik, 0);
Serial.print("e");
Serial.println(load_volt);
Serial.print("f");
Serial.println(headingDegrees);
}

```

LAMPIRAN F

Standard Operating Procedure (SOP) to Operating Buoyant Boat

To operate Buoyant Boat, there are some step you must to do. The steps is below,

1. *Make sure that your battery is in full charged or over 80%.*
2. *Bring Buoyant Boat to the lake or wide water.*
3. *Turn on Buoyant Boat.*
4. *Turn on Remote Control (RC).*
5. *Set SwG Switch in position 1 (facing near human) to select manual control mode and then control Buoyant Boat as you want with controlling Throttle and Aille.*
6. *Set SwG Switch to position 2 (middle) to select auto mode, then Buoyant Boat will be follow the path.*
7. *Set SwG Switch to position 3 to select loiter mode, then set SwE Switch in position 1 (facing near human) to update location, and change the position to position 2 (middle), so Buoyant Boat will do loiter mode movement.*



(Source: Radiolink AT9 (Dsss) Instruction Manual,
<http://www.radiolink.com.cn>)

More information: SwG Switch to select mode (manual, auto, and loiter), SwE Switch to lock location while in loiter mode, then Throttle and Aille stick is used as control manual.

.....*Halaman ini sengaja dikosongkan*.....

LAMPIRAN G
Dokumentasi Kegiatan Tugas Akhir



.....*Halaman ini sengaja dikosongkan*.....

BIODATA PENULIS



Penulis dilahirkan di Lumajang pada tanggal 25 Desember 1996. Sebagai anak kedua dari empat bersaudara, penulis mengawali kegiatan pendidikan formal di SD Negeri 03 Kalipenggung, yang kemudian dilanjutkan di SMP Negeri 01 Randuagung, SMA Negeri 03 Lumajang dan pada tahun 2014 penulis diterima sebagai mahasiswa di Departemen Teknik Elektro ITS dengan Beasiswa Bidikmisi. Selain Beasiswa Bidikmisi, penulis juga memperoleh Beasiswa lain, yaitu

Beastudi Etos, Beasiswa Dataprint, dan Beasiswa Ika Elits. Selama menjalani perkuliahan, penulis aktif dalam kepanitian, kegiatan sosial, organisasi, dan kompetisi, mulai dari tingkat jurusan hingga tingkat nasional. Selain itu, penulis juga turut berpartisipasi sebagai asisten praktikum di bidang studi elektronika.

Email : tamambadrut13@gmail.com
Hp/WA : 082245479239
Instagram : @tamambadrut13
Facebook : Badrut Tamam
Line : badrut25
LinkedIn : Badrut Tamam